ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR - 141501

RANCANG BANGUN PERANGKAT LUNAK TEENSTAGRAM UNTUK MENGELOMPOKKAN TOPIK CAPTION AKUN INSTAGRAM SISWA SEKOLAH MENENGAH PERTAMA DI SURABAYA

TEENSTAGRAM APPLICATION FOR TOPIC CAPTION CLASSIFICATION FROM THE INSTAGRAM ACCOUNTS OF JUNIOR HIGH SCHOOL STUDENTS IN SURABAYA

TETHA VALIANTA
NRP 5213100055

Dosen Pembimbing
Nur Aini Rakhmawati, S.Kom., M.Sc.Eng., Ph.D
Irmasari Hafidz, S.Kom, M.Sc

JURUSAN SISTEM INFORMASI
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember
Surabaya, 2017

Halaman ini sengaja dikosongkan

TUGAS AKHIR - 141501

RANCANG BANGUN PERANGKAT LUNAK TEENSTAGRAM UNTUK MENGELOMPOKKAN TOPIK CAPTION AKUN INSTAGRAM SISWA SEKOLAH MENENGAH PERTAMA DI SURABAYA

TETHA VALIANTA
NRP 5213100055

Dosen Pembimbing
Nur Aini Rakhmawati, S.Kom., M.Sc.Eng., Ph.D
Irmasari Hafidz, S.Kom, M.Sc

JURUSAN SISTEM INFORMASI
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember
Surabaya, 2017

Halaman ini sengaja dikosongkan

UNDERGRADUATE THESIS - 141501

TEENSTAGRAM APPLICATION FOR TOPIC CAPTION CLASSIFICATION FROM THE INSTAGRAM ACCOUNTS OF JUNIOR HIGH SCHOOL STUDENTS IN SURABAYA

TETHA VALIANTA
NRP 5213100055

Supervisor
Nur Aini Rakhmawati, S.Kom., M.Sc.Eng., Ph.D

Irmasari Hafidz, S.Kom, M.Sc

DEPARTMENT OF INFORMATION SYSTEM
Faculty of Information Technology
Institut Teknologi Sepuluh Nopember
Surabaya, 2017

Halaman ini sengaja dikosongkan

LEMBAR PENGESAHAN

RANCANG BANGUN PERANGKAT LUNAK TEENSTAGRAM UNTUK MENGELOMPOKKAN TOPIK CAPTION AKUN INSTAGRAM SISWA SEKOLAH MENENGAH PERTAMA DI SURABAYA

TUGAS AKHIR

Diajukan Guna Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Bidang Studi Analisa Data dan Diseminasi Informasi
Program Studi S1 Departemen Sistem Informasi
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh :

TETHA VALIANTA
NRP: 5213100055

Surabaya, September 2017

KEPALA
DEPARTEMEN SISTEM INFORMASI

Dr. Ir. Aris Tjahyanto, M.Kom.
NIP. 19650310 199102 1 001

Halaman ini sengaja dikosongkan

LEMBAR PERSETUJUAN

RANCANG BANGUN PERANGKAT LUNAK TEENSTAGRAM UNTUK MENGELOMPOKKAN TOPIK CAPTION AKUN INSTAGRAM SISWA SEKOLAH MENENGAH PERTAMA DI SURABAYA

TUGAS AKHIR

Diajukan Guna Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Bidang Studi Analisa Data dan Diseminasi Informasi
Program Studi S1 Departemen Sistem Informasi
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh :

TETHA VALIANTA
NRP: 5213100055

Disetujui Tim Penguji: Tanggal Ujian: 7 Juli 2017
Periode Wisuda: September 2017

Nur Aini Rakhmawati, S.Kom., M.Sc.Eng., Ph.D (Pembimbing 1)

Irmasari Hafidz, S.Kom, M.Sc (Pembimbing 2)

Faizal Johan Atletiko, S.Kom., M.T (Penguji 1)

Radityo Prasetianto Wibowo, S.Kom, M.Kom (Penguji 2)

Halaman ini sengaja dikosongkan

RANCANG BANGUN PERANGKAT LUNAK TEENSTAGRAM UNTUK MENGELOMPOKKAN TOPIK CAPTION AKUN INSTAGRAM SISWA SEKOLAH MENENGAH PERTAMA DI SURABAYA

Nama	: TETHA VALIANTA
NRP	: 5213100055
Jurusan	: Sistem Informasi FTIf
Pembimbing I	: Nur Aini Rakhmawati, S.Kom., M.Sc.Eng., Ph.D
Pembimbing II	: Irmasari Hafidz, S.Kom, M.Sc

Abstrak

Menurut Kementerian Komunikasi dan Informatika RI 30 juta anak-anak dan remaja di Indonesia merupakan pengguna internet, dan media sosial menjadi pilihan utama sarana komunikasi yang mereka gemari. Menurut Asosiasi Penyelenggaraan Jasa Internet Indonesia (APJII), Instagram merupakan media sosial yang berhasil mengambil tempat di hati penggunanya, terbukti dengan kenaikan yang signifikan sejak pertama diluncurkan pada Oktober 2010, tercatat ada 500 juta pengguna aktif dunia di tahun 2016, dan di Indonesia tercatat sebanyak 19,9 juta atau sebesar (15 persen) pengguna. Menurut KOMINFO tingginya animo pengguna Instagram khususnya usia remaja tidak diimbangi dengan pengawasan orang tua dikarenakan faktor orang tua yang mengaku kewalahan untuk mengawasi putra-putrinya dalam berekspresi di media sosial. Dari fenomena tersebut dibutuhkan sebuah platform yang mampu memberikan informasi visual terhadap aktivitas remaja dalam hal berekspresi di media sosial Instagram, dengan cara melakukan analisis topic modelling terhadap perilaku atau kebiasaan remaja ketika upload gambar disertai dengan caption tertentu, menggunakan metode Latent Dirichlet Allocation atau yang akrab disebut dengan LDA. Penelitian ini dikhususkan untuk menganalisa data

caption akun Instagram siswa SMP di Surabaya, setelah data akun dan data caption didapatkan serta dianalisa menggunakan LDA, kemudian dilakukan visualisasi terhadap topik atau kategori aktivitas siswa berdasarkan captionnya. Melalui proses pembuatan model menggunakan LDA telah didapatkan hasil terbaik berupa 2 topik. Adapun 2 topik tersebut dapat dikatakan baik karena memiliki nilai perplexity yang kecil, yang artinya model yang dibuat memiliki tingkat kesesuaian yang bagus. Dua topik yang terbentuk pada proses ini diterjemahkan ke dalam dua kategori, yakni ***edukasi*** *dan* ***interaksi****. Berdasarkan hasil prediksi pelabelan data, didapatkan informasi bahwa model topik didominasi oleh kategori* ***Edukasi*** *dengan jumlah data* ***3940****, sedangkan kategori interaksi memiliki jumlah 724.*

Kata Kunci: Instagram, Caption, Topik, SMP, LDA, Latent Dirichlet Allocation

TEENSTAGRAM APPLICATION FOR TOPIC CAPTION CLASSIFICATION FROM THE INSTAGRAM ACCOUNTS OF JUNIOR HIGH SCHOOL STUDENTS IN SURABAYA

Name	: TETHA VALIANTA
NRP	: 5213100055
Major	: Information System FTIf
Supervisor I	: Nur Aini Rakhmawati, S.Kom., M.Sc.Eng., Ph.D
Supervisor II	: Irmasari Hafidz, S.Kom, M.Sc

Abstract

According to the Ministry of Communications and Information Technology, 30 million children and adolescents in Indonesia are internet users, and social media becomes the main choice of communication means they enjoy. According to the Indonesian Internet Services Association (APJII), textit Instagram is a social media that successfully takes its place in the hearts of its users, as evidenced by the significant increase since it was first launched in October 2010, there are 500 million active users worldwide by 2016, In Indonesia there were 19.9 million or 15 percent of users. According KOMINFO high user interest textit Instagram especially adolescents are not matched with parental supervision due to parents who claim to be overwhelmed to supervise his sons and daughters in expression in social media. From that phenomenon it takes a textit platform that is able to provide visual information on youth activities in terms of expression in social media textit Instagram, by analyzing topic modeling on teenage behavior or habits when textit upload The image is accompanied by a certain textit caption, using the textit Latent Dirichlet Allocation or the so-called textit LDA method. This study was devoted to analyzing the account account data of SMP students in Surabaya, after account data and data caption were obtained and analyzed using textit LDA, then visualization of

the topic or category of student activity based on textit Caption. Through the process of modeling using it LDA has got the best results in 2 topics. The two topics can be said to be good because it has a small it perplexity value, which means the model created has a good level of conformity. Two topics formed in this process are translated into two categories: textbf educational and textbf interaction. Based on predicted data labeling results, it is found that the topic model is dominated by the textbf Education category with the data number textbf 3940, while the interaction category has the number 724.
Keywords: Repository, Software, Linked Data, DBpedia

KATA PENGANTAR

Puji syukur penulis haturkan ke hadirat Tuhan YME yang telah memberikan anugerah dan tuntunan kepada penulis sehingga penulis dapat menyelesaikan tugas akhir dengan judul *"RANCANG BANGUN PERANGKAT LUNAK TEENSTAGRAM UNTUK MENGELOMPOKKAN TOPIK CAPTION AKUN INSTAGRAM SISWA SEKOLAH MENENGAH PERTAMA DI SURABAYA"* sebagai salah satu syarat kelulusan pada Jurusan Sistem Informasi, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember Surabaya. Penyusunan tugas akhir ini senantiasa mendapatkan dukungan dari berbagai pihak baik dalam bentuk doa, motivasi, semangat, kritik, saran dan berbagai bantuan lainnya. Untuk itu, secara khusus penulis akan menyampaikan ucapan terima kasih yang sedalam-dalamnya kepada:

1. Segenap keluarga besar terutama kedua orang tua penulis, Bapak Djoko Wahyudianto S.Si, dan Ibu Yekti Prihatin S.Si yang senantiasa mendoakan, memberikan motivasi dan semangat, sehingga penulis mampu menyelesaikan pendidikan S1 ini dengan baik.
2. Bapak Dr. Ir. Aris Tjahyanto, M.Kom., selaku Ketua Jurusan Sistem Informasi ITS, Bapak Nisfu Asrul Sani, S.Kom, M.Sc selaku KaProdi S1 Sistem Informasi ITS serta seluruh dosen pengajar beserta staf dan karyawan di Jurusan Sistem Informasi, FTIF ITS Surabaya selama penulis menjalani kuliah
3. Ibu Nur Aini Rakhmawati, S.Kom., M.Sc., Eng. Ph.D dan Ibu Irmasari Hafidz, S.Kom, M.Sc selaku dosen pembimbing yang telah banyak meluangkan waktu untuk membimbing, mengarahkan, dan mendukung dengan memberikan ilmu, petunjuk, dan motivasi dalam penyelesaian Tugas Akhir
4. Bapak Bekti Cahyo Hidayanto, S.Si., M.Kom sebagai dosen wali penulis selama menempuh pendidikan di Jurusan Sistem

Informasi.

5. Bapak Faizal Johan Atletiko, S.Kom., M.T dan Bapak Radityo Prasetianto Wibowo, S.Kom, M.Kom selaku dosen penguji yang telah memberikan kritik, saran, dan masukan yang dapat menyempurnakan Tugas Akhir ini.
6. Teman-teman Sistem Informasi angkatan 2013 (13ELTRANIS) yang senantiasa menemani dan memberikan motivasi bagi penulis selama perkuliahan hingga dapat menyelesaikan tugas akhir.
7. Rekan-rekan organisasi Himpunan Mahasiswa Sistem Informasi Kabinet Muda Berkarya serta staff 2014 (Lil) yang telah memberikan doa, semangat, perhatian serta motivasi.
8. ”Kapten Harun Rizal” CEO intip.in yang telah banyak memberikan ilmu dan membantu penulis.
9. Rekan-rekan ”Sahabat Sambat”, ”Sekitar Kita”, ”UKM PENALARAN”, ”Pohong” atas kebersamaan dan kenangan yang selalu berkesan.
10. Serta seluruh pihak-pihak lain yang tidak dapat disebutkan-satu per satu yang telah banyak membantu penulis selama perkuliahan hingga dapat menyelesaikan tugas akhir ini.

Terima kasih atas segala bantuan, dukungan, serta doanya. Semoga Tuhan YME senantiasa melimpahkan anugerah serta membalas kebaikan yang telah diberikan kepada penulis.

Penulis menyadari bahwa masih terdapat kekurangan dalam penyusunan tugas akhir ini, oleh karena itu penulis mengharapkan saran dan kritik yang membangun demi kebaikan penulis dan tugas akhir ini. Akhir kata, penulis berharap bahwa tugas akhir ini dapat memberikan kebermanfaatan

DAFTAR ISI

DAFTAR TABEL

DAFTAR GAMBAR

DAFTAR KODE

Halaman ini sengaja dikosongkan

BAB 1

PENDAHULUAN

Pada bab pendahuluan akan diuraikan proses identifikasi masalah penelitian yang meliputi latar belakang masalah, perumusan masalah, batasan masalah, tujuan tugas akhir, manfaat kegiatan tugas akhir dan relevansi terhadap pengerjaan tugas akhir. Berdasarkan uraian pada bab ini, harapannya gambaran umum permasalahan dan pemecahan masalah pada tugas akhir dapat dipahami.

1.1 Latar Belakang

Menurut Kementerian Komunikasi dan Informatika RI 30 juta anak-anak dan remaja di Indonesia merupakan pengguna internet, dan media sosial menjadi pilihan utama sarana komunikasi bagi mereka, telah tercatat sebesar (73 persen) remaja mengakses internet menggunakan smartphone mereka, dan (4 persen) menggunakan tablet [14]. Selaras dengan perkembangan internet, penggunaan media sosial di kalangan remaja telah menjadi kebiasaan kehidupan sehari-hari mereka. Menurut studi ini ditemukan 98 persen dari anak-anak dan remaja yang disurvei tahu tentang internet dan 79,5 persen diantaranya adalah pengguna aktif media sosial. Faktor yang mendasari para remaja mengakses internet dan media sosial menurut riset Kementerian Komunikasi dan Informatika RI dan UNICEF ada tiga, yakni untuk mencari informasi, terhubung dengan teman (lama ataupun baru) dan untuk hiburan semata [14].

Tingginya rasa ingin tahu remaja terhadap hal baru tentunya mendorong orang tua untuk berperan aktif dalam mengawasi putra-putrinya pada saat mengakses internet, adapun menurut penelitian yang di-

lakukan dilakukan oleh Kominfo dan UNICEF mengatakan bahwa para orang tua merasa kewalahan dalam memantau atau mendampingi putra-putrinya dalam mengakses internet maupun media sosial, khususnya media sosial yang marak diakses oleh remaja, seperti *Facebook, Twitter, Instagram*. Tingkat penetrasi media sosial di kalangan remaja sendiri, khususnya di wilayah perkotaan, Surabaya, menurut penelitian Pengaruh dan Pola Aktivitas Penggunaan Internet serta Media Sosial pada Siswa SMPN 52 Surabaya didominasi oleh *Facebook* dengan prosentase pengguna (14 persen) dari keseluruhan pengguna, kemudian, disusul dengan *WhatsApp, Twitter, Facebook Messenger, Google+, LinkedIn, Instagram, Skype, Pinterest* dan urutan terakhir ditempati *LINE* dengan presentase (6 persen)[21]. Dilansir dalam survei yang dilakukan oleh Asosiasi Penyelenggaraan Jasa Internet Indonesia [2] menyatakan bahwa pengguna *Instagram* mengalami kenaikan yang signifikan sejak pertama diluncurkan pada Oktober 2010, tercatat ada 500 juta pengguna aktif dunia di tahun 2016, dan di Indonesia tercatat sebanyak 19,9 juta atau sebesar (15 persen) pengguna. Seiring meningkatnya jumlah pengguna media sosial khususnya *instagram*, dilakukan proses percobaan pencarian kata kunci "media sosial remaja" menggukan mesin pencari Google, dan ditemukan sebanyak 785,000 hasil pencarian, yang mayoritas menunjukkan dampak-dampak dari penggunaan media sosial di kalangan remaja, mulai dari pergaulan, kenalakan remaja hingga dampak negatif yang tergolong fatal, seperti pertikaian sesama remaja.

Dari fenomena yang terjadi terkait penggunaan media sosial di kalangan remaja dibutuhkan sebuah *platform* yang mampu memberikan informasi visual terhadap aktivitas remaja dalam hal berekspresi di media sosial *Instagram*, dengan cara melakukan analisis topic modelling terhadap perilaku atau kebiasaan remaja ketika *upload* gambar disertai dengan *caption* tertentu, menggunakan metode *Latent Dirichlet Allocation* atau yang akrab disebut dengan *LDA*. Pe-

nelitian ini dikhususkan untuk menganalisa data caption akun Instagram siswa SMP dari 18 sekolah di Surabaya, yang menjadi target untuk kelas mata kuliah Etika Profesi di Jurusan Sistem Informasi Institut Teknologi Sepuluh Nopember Surabaya. Setelah data akun dan data caption didapatkan serta dianalisa menggunakan *LDA*, kemudian dilakukan visualisasi terhadap topik atau kategori aktivitas siswa berdasarkan *caption*nya. Melalui penelitian ini diharapkan mampu memberikan informasi kepada para orang tua dan guru untuk lebih peka dan mampu mengarahkan putra-putrinya agar dapat bertanggung jawab dalam hal berekspresi di media sosial, sehingga pelanggaran serta dampak negatif yang dikhawatirkan dari pengaruh media sosial dapat diminimalisir.

1.2 Perumusan Masalah

Berdasarkan uraian latar belakang, maka rumusan permasalahan yang menjadi fokus dan akan diselesaikan dalam Tugas Akhir ini antara lain :

1. Bagaimana cara mengakuisisi data akun dan *caption* anak SMP di Surabaya melalui media sosial *instagram*?
2. Bagaimana cara merancang *platform* yang mampu melakukan visualisasi data topik pembicaraan pelajar SMP ke dalam sebuah gambar atau *dashboard*?
3. Bagaimana cara mengelompokkan topik caption atau caption instagram dari akun siswa SMP?
4. Bagaimana melakukan pemodelan serta pelabelan data secara berkelanjutan pada periode tertentu?

1.3 Batasan Masalah

Dari permasalahan yang disebutkan di atas, batasan masalah dalam tugas akhir ini adalah :

1. Studi kasus yang digunakan pada penelitian ini hanya meliputi 18 Sekolah Menegah Pertama di wilayah Surabaya.
2. Pengambilan data dalam tugas akhir ini dilakukan hanya dari data *caption account instagram* pelajar SMP di Surabaya tanpa mengambil data di kolom komentar.
3. Proses pembuatan model dengan LDA digunakan hanya sebatas untuk visualisasi dan mengetahui topik pembicaraan akun siswa SMP.

1.4 Tujuan Tugas Akhir

Berdasarkan hasil perumusan masalah dan batasan masalah yang telah disebutkan sebelumnya, maka tujuan yang dicapai dari tugas akhir ini adalah untuk menciptakan sebuah *platform* yang mampu mengklasifikasi dan memvisualkan perilaku atau aktivitas pelajar SMP di media sosial *instagram* guna mendorong para orang tua atau guru untuk lebih memperhatikan putra-putrinya dalam hal berekspresi di media sosial. Sehingga dengan adanya *platform* ini diharapkan para pelajar SMP di Surabaya mampu berekspresi di media sosial dengan lebih bertanggung jawab.

1.5 Manfaat Tugas Akhir

Manfaat yang diharapkan dapat diperoleh dari tugas akhir ini adalah:

1. Memfasilitasi orang tua dan guru dalam mengawasi pergaulan pelajar SMP di Surabaya.
2. Memfasilitasi mahasiswa, khususnya Jurusan Sistem Informasi untuk mempelajari *social network analysis*
3. Menyediakan data yang dapat digunakan sebagai acuan untuk menentukan tindakan lebih lanjut dalam hal kampanye penggunaan *smartphone dan social media* yang bertanggung jawab kepada pelajar di wilayah Surabaya.

1.6 Relevansi Tugas Akhir

Tugas akhir ini berkaitan dengan mata kuliah Pemrograman Berbasis Web, Analisa dan Desain Perangkat Lunak dan Konstruksi Pengembangan Perangkat Lunak, Penggalian Data Analitika Bisnis, Pemrograman Integratif, dan Etika Profesi.

Halaman ini sengaja dikosongkan

BAB 2

TINJAUAN PUSTAKA

Bab ini akan menjelaskan mengenai penelitian sebelumnya dan dasar teori yang dijadikan acuan atau landasan dalam pengerjaan tugas akhir ini. Landasan teori akan memberikan gambaran secara umum dari landasan penjabaran tugas akhir ini.

2.1 Penelitian Sebelumnya

Pada subbab ini dijelaskan tentang referensi penelitian yang berkaitan dengan tugas akhir. Pada bagian ini memaparkan acuan penelitian sebelumnya yang digunakan oleh penulis dalam melakukan penelitiannya.

1. Penelitian pertama berjudul *"A First Analysis of Instagram Photo Content and User Types"* oleh Yuheng Hu, Lydia Manikonda, dan Subbarao Kambhampati [11]. Dalam penelitian tersebut dilakukan pengumpulan data (profil, foto, followers pengguna, keterangan dan tag terkait dengan foto) pengguna Instagram menggunakan API Instagram menggunakan metode random sample atau menggunakan sampel acak dari pengguna Instagram. Peneliti menggunakan *computer vision techniques* menguji konten foto yang didapatkan, kemudian peneliti mengidentifikasi berbagai jenis pengguna aktif di Instagram menggunakan metode clustering. Hasil dari penelitian tersebut adalah menunjukkan bahwa sebagian besar terdapat 8 jenis kategori foto pada social media Instagram, yang berasal 5 jenis users. Menurut analisa menunjukkan bahwa tidak ada hubungan langsung antara jumlah pengikut dan tipe

pengguna dicirikan dalam hal konten foto yang dipostingnya, melalui uji signifikansi statistic.

2. Penelitian kedua berjudul "Analyzing User Activities, Demographics, Social Network Structure and User-Generated Content on Instagram" oleh Lydia Manikonda Yuheng Hu Subbarao Kambhampati [17]. Dalam makalah ini peneliti menganalisis konten, fitur geografis dan sifat jaringan sosial dari media sosial Instagram. Peneliti mengumpulkan dataset pengguna instagram yang diambil dari sample acak, kemudian peneliti melakukan *social network analysis* menggunakan homophily, reciprocity, clustering coefficient. Hasil dari penelitian ini ditemukan beberapa fakta, yakni, 1) Instagram memiliki sifat *social network* yang sangat berbeda dari media sosial populer lainnya seperti Twitter dan Flickr, Instagram tergolong dalam *asymmetric social awareness platform*, 2) User biasanya melakukan *posting* melalui akunnya setiap seminggu sekali, dan 3) mayoritas orang selalu ingin berbagi lokasi dengan teman-temannya, makalah ini mengklaim bahwa penelitian ini adalah penelitian pertama yang melakukan analisa mendalam terkait aktivitas pengguna, demografi, struktur jaringan sosial dan *user-generated content* di Instagram.
3. Penelitian ketiga berjudul *C*lassification Via Clustering for Predicting Final Marks Based on Student Participation in Forums" oleh M.I. López, J.M Luna, C. Romero, S. Ventura [16]. Dalam penelitian ini dilakukan klasifikasi melalui pendekatan clustering untuk memprediksi final mark di sebuah universitas berdasarkan data forum. Tujuannya adalah untuk menentukan apakah partisipasi siswa dalam forum dapat menjadi prediktor yang baik bagi final mark dalam perkuliahan dan untuk membuktikan apakah klasifikasi melalui clustering dapat menghasilkan akurasi yang sama dengan algoritma klasifikasi tradisional. Percobaan dilakukan meng-

gunakan data real dari mahasiswa tahun pertama. Beberapa algoritma clustering digunakan dan dibandingkan dengan algoritma klasifikasi tradisional dalam memprediksi apakah siswa lulus atau gagal dalam proses perkuliahan berdasarkan data dari forum Moodle mereka. Hasil penelitian menunjukkan bahwa partisipasi siswa dalam forum dapat menjadi prediktor yang baik bagi final mark dalam perkuliahan dan Expectation-Maximisation (EM) hasil algoritma clustering menunjukkan hasil akurasi yang mirip dengan algoritma klasifikasi tradisional.

4. Penelitian keempat berjudul "Detection of Cyberbullying Incidents on the Instagram Social Network" oleh Homa Hosseinmardi, Sabrina Arredondo Mattson, Rahat Ibn Rafiq, Richard Han, Qin Lv, Shivakant Mishra [10]. Dalam penelitian ini dilakukan pengumpulan satu set sampel Instagram data yang terdiri dari gambar dan komentar, kemudian data terkumpul, dilakukan proses pemberian label untuk cyberbullying menggunakan *labelers* manusia melalui situs web *Crowdflower*. Melalui data yang telah dilabelkan didapatkan hasil bahwa sekitar (48 persen) *Instagram media sessions* tidak dianggap sebagai cyberbullying atas dasar kriteria lima labelers, ada korelasi yang kuat antara kekuatan dukungan untuk cyberbullying label dan jumlah komentar teks serta properti temporal jumlah komentar yang diposting dalam waktu satu jam dalam *Instagram media sessions* dan peneliti menggunakan Linear SVM classifier yang secara signifikan mampu meningkatkan akurasi identifikasi cyberbullying ke (87 persen) dengan memasukkan fitur multi-modal dari teks, gambar, dan meta data dari data terlabel.
5. Penelitian kelima berjudul *"A Text mining Research Based On LDA Topic Modelling"* oleh Zhou Tong dan Haiyi Zhang, paper ini memperkenalkan Text mining dengan LDA topic modelling sebagai metodenya, dimana eksperimen dilakukan

pada dua tipe dokumen, yaitu artikel Wikipedia dan tweet dari pengguna Twitter. Garis besar penelitian ini membahas tentang gambaran umum text mining dengan metode LDA, pre-processing, model training dan hasil analisa [23].

6. Penelitian keenam berjudul *"Software Framework for Topic modelling with Large Corpora"* oleh Radim Rehurek dan Petr Sojka [19]. Penelitian ini menggunakan metode Vector Space Model (VSM) yaitu Latent Semantik Analysis (LSA) dan Latent Dirichlet Allocation (LDA) dengan framework Gensim dan Bahasa Python. Vector Space Model (VSM) adalah paradigma dalam modelling yang terbukti dan ampuh dalam Natural Language Processing, di mana dokumen direpresentasikan sebagai vektor dalam ruang berdimensi tinggi. Metode yang termasuk dalam VSM cukup beragam, sehingga algoritma yang diterapkan juga beragam. Dengan demikian, diperlukan suatu framework untuk memberikan arahan sebagai solusi dari adanya practical gap antara model matematis, algoritma dan source code. Penelitian ini mengajukan adanya framework yang mencakup aspek Corpus size independence, Intuitive API, Easy deployment, Cover popular algorithms dengan menggunakan bahasa Python.

2.2 Dasar teori

2.2.1 Instagram

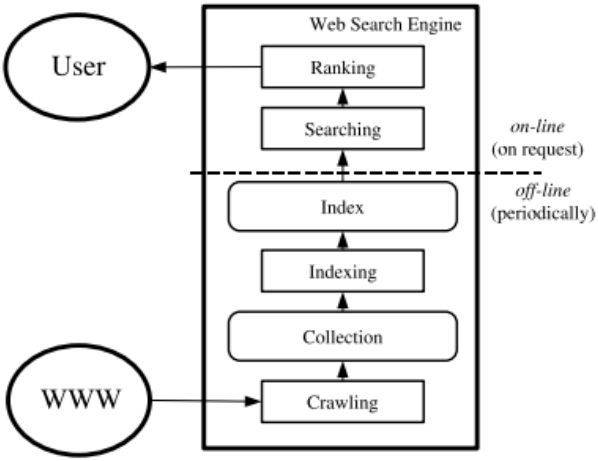
Instagram adalah situs online yang digunakan untuk berbagi foto, layanan Instagram memungkinkan penggunanya untuk berbagi foto dan video secara publik, serta dapat juga dibagikan melalui berbagai platform jejaring sosial lainnya, seperti Facebook, Twitter, Tumblr, dan Flickr. Instagram juga menyediakan layanan API (Application Programming Interface) [12], dalam praktiknya Insta-

gram API biasanya digunakan untuk memudahkan pengambilan data berupa *caption*, *hashtag*, ataupun foto pada akun pengguna *Instagram* untuk kebutuhan analisa tren, survey atau penelitian tertentu [8]. Dalam Instagram ada beberapa batasan yang perlu diperhatikan [13]:

1. Batas jumlah orang yang dapat diikuti *following* oleh satu akun adalah 7500
2. Batas untuk melakukan like sebanyak 350 per hari
3. Batas tagar (*hashtag* yang diperbolehkan adalah 30 hashtag per post
4. Batas karakter untuk Biodata adalah 150 karakter dan
5. Batas karakter untuk Caption adalah 2200 karakter

2.2.2 Google Maps

Google Maps adalah layanan aplikasi peta online yang disediakan oleh Google secara gratis. Layanan peta Google Maps secara resmi dapat diakses melalui situs http://maps.google.com. Pada situs tersebut dapat dilihat informasi geografis pada hampir semua permukaan di bumi kecuali daerah kutub utara dan selatan. Layanan ini dibuat sangat interaktif, karena di dalamnya peta dapat digeser sesuai keinginan pengguna, mengubah level zoom, serta mengubah tampilan jenis peta. Google Maps mempunyai banyak fasilitas yang dapat dipergunakan misalnya pencarian lokasi dengan memasukkan kata kunci, kata kunci yang dimaksud seperti nama tempat, kota, atau jalan, fasilitas lainnya yaitu perhitungan rute perjalanan dari satu tempat ke tempat lainnya. BatchGeo adalah salah satu fitur yang memanfaatkan google maps API yang digunakan untuk pemetaan alamat, kemudian dibagikan ke dalam informasi berupa peta. Batchgeo dapat ditanamkan dalam website berbentuk HTML, dari hasil data yang ingin dipetakan sesuai keinginan. Batchgeo ju-



Gambar 2.1: Proses Kerja *Web Crawler* [7]

ga menyediakan KML ekspor sehingga peta yang dihasilkan dapat dilihat menggunakan Google Earth, Google Maps, ArcMap, atau sejumlah klien pemetaan populer lainnya.

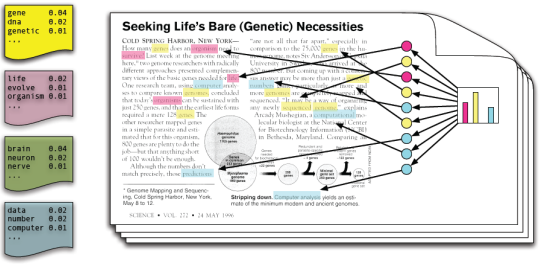
2.2.3 Crawler

Web Crawler atau juga dapat dikenal sebagai robot, atau laba-laba. *Web Crawler* berbentuk sebagai program atau script dimana dengan metode tertentu program tersebut dapat melakukan proses pemindaian ke semua halaman-halaman web untuk membuat indeks dari data yang menjadi tujuan pencarian [9]. Pada gambar 2.1, secara umum *Web Crawler* bekerja dalam dua bagian utama, yaitu bagian offline dan bagian online. Bagian offline secara periodik dieksekusi oleh mesin pencari, dan pada proses tersebut, *crawler* mengunduh beberapa bagian tertentu dari web untuk membentuk sekumpulan

halaman, yang nantinya dapat disatukan dan menjadi *searchable index*. Selanjutnya, di bagian kedua, yaitu bagian online, dieksekusi setiap kali ada permintaan pengguna yang dieksekusi, dan menggunakan index untuk memilih beberapa kandidat dokumen yang telah diurutkan menurut perkiraan seberapa penting dokumen tersebut dengan keinginan yang diharapkan pengguna [7].

2.2.4 Topic Modelling

Menurut Blei kosep topic modelling terdiri dari entitas-entitas yaitu ”kata”, ”dokumen”, dan ”corpora”. ”Kata” merupakan unit dasar dari data diskrit dalam dokumen, didefinisikan sebagai item dari kosakata yang diberi indeks untuk setiap kata unik pada dokumen. Sedangkan ”dokumen” merupakan susunan N kata. Corpus adalah kumpulan M dokumen dan corpora merupakan bentuk jamak dari corpus. Sedangkan ”topic” adalah distribusi dari beberapa kosakata yang bersifat tetap. Dengan kata lain, setiap dokumen dalam corpus mengandung proporsi tertentu dari topik-topik yang dibahas sesuai kata-kata yang terdapat di dalamnya [5]. Dasar dari topic modeling adalah bahwa sebuah topik terdiri dari kata-kata tertentu yang menyusun topik tersebut, dan dalam satu dokumen memiliki kemungkinan terdiri dari beberapa topik dengan probabilitas masing-masing. Keterbatasan kemampuan manusia dalam memahami topik, distribusi topik per-dokumen, dan penggolongan setiap kata pada topik per-dokumen mendorong adanya topic modelling yang bertujuan untuk menemukan topik dan kata-kata yang terdapat pada topik tertentu [4]. Konsep topic modelling menurut Blei, ditunjukkan pada gambar 2.2 Topic modelling merupakan algoritma yang bertujuan untuk menemukan topik yang tersembunyi dari rangkaian kata dalam dokumen yang tidak terstruktur. Algoritma topic modelling menganalisis kata-kata dari teks asli untuk menemukan topik yang berada diantara teks tersebut, bagaimana topik

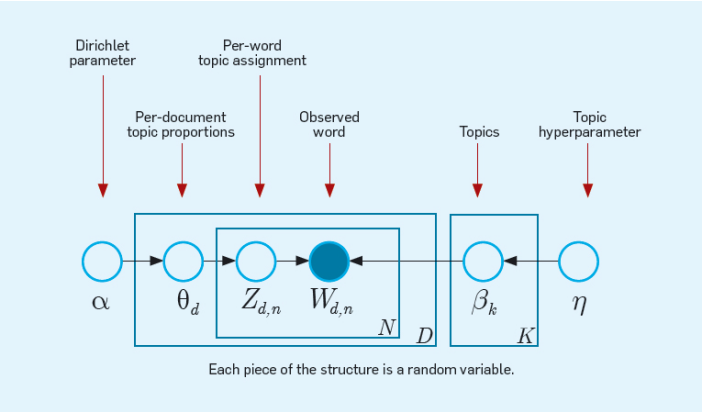


Gambar 2.2: Konsep *Topic Modelling [4]*

topik saling terhubung satu sama lain, dan bagaimana tema-tema tersebut dapat berubah dari waktu ke waktu, sehingga dapat dikembangkan untuk pencarian, ataupun meringkas teks yang terdapat dalam dokumen [5]. Algoritma pemodelan topik tidak memerlukan penjelasan sebelumnya atau pelabelan karena topik dokumen muncul dari analisis teks asli. Pada umumnya, topic modelling diaplikasikan pada jumlah dokumen yang sangat besar dan dapat mengelola dokumen-dokumen individual sesuai topik yang ditemukan, sehingga topic modelling memungkinkan kita untuk mengatur dan meringkas arsip elektronik pada skala yang tidak mungkin dengan penjelasan manusia secara manual [4].

2.2.5 Latent dirichlet allocation

Latent Dirichlet Allocation (LDA) merupakan sebuah metode statistika yang digunakan sebagai model untuk menganalisis suatu dokumen. LDA dipilih sebagai salah satu metode dalam melakukan analisis pada dokumen yang berukuran sangat besar. LDA dapat digunakan untuk meringkas, melakukan klasterisasi, menghubungkan maupun memproses data yang sangat besar karena LDA menghasilkan daftar topik yang diberi bobot untuk masing-masing



Gambar 2.3: Distribusi Topik LDA [4]

dokumen [6]. Adapun metode distribusi yang digunakan untuk mendapatkan distribusi topik per-dokumen disebut distribusi Dirichlet, kemudian hasil dari Dirichlet digunakan untuk mengalokasikan kata-kata pada dokumen untuk topik yang berbeda. Dalam LDA, dokumen-dokumen merupakan objek yang dapat diamati, sedangkan topik, distribusi topik per-dokumen, penggolongan setiap kata pada topik per-dokumen merupakan struktur tersembunyi yang tidak dapat diamati secara manual oleh manusia, oleh sebab itu algoritma ini dinamakan Latent Dirichlet Allocation (LDA) [4]. Blei merepresentasikan metode LDA sebagai model probabilistic secara visual seperti pada gambar 2.3: Sesuai visualisasi model 2.3, parameter α dan β merupakan parameter distribusi topik yang berada pada tingkatan *corpus*, yaitu kumpulan dari M dokumen. Parameter α digunakan dalam menentukan distribusi topik dalam dokumen, semakin besar nilai alpha dalam suatu dokumen, menandakan semakin banyak campuran topik yang dibahas dalam dokumen, sedangkan semakin rendah nilai α menunjukkan bahwa dalam dokumen hanya membahas sedikit topik tertentu, Z merepresentasikan topik dari kata tertentu pada sebuah dokumen. Parameter β digu-

Gambar 2.4: Ilustrasi LDA menurut Blei [4]

nakan untuk menentukan distribusi kata dalam topik. Semakin tinggi nilai β, maka semakin banyak kata-kata yang ada di dalam topik, sedangkan semakin kecil nilai β, maka semakin sedikit kata-kata yang ada di dalam topik sehingga topik tersebut dapat dikatakan lebih spesifik. Variabel θ_m adalah variabel yang berada di tingkat dokumen (M). Variabel θ merepresentasikan distribusi topik untuk dokumen tertentu. Semakin tinggi nilai θ, maka semakin banyak topik yang ada di dalam dokumen, sedangkan semakin kecil nilai θ, maka dapat dikatakan dokumen tersebut semakin spesifik pada topik tertentu [5]. Ide dasar dari LDA adalah bahwa dalam dokumen, merepresentasikan campuran topik secara acak, dimana setiap topik digolongkan berdasarkan distribusi antar kata. Sebagai salah satu contoh dari Blei, distribusi topik yang ditampilkan dengan kumpulan kata-kata pada dokumen ditunjukkan dengan gambar 2.4 Secara umum, LDA bekerja dengan masukan dokumen-dokumen individual dan beberapa parameter, untuk menghasilkan luaran berupa model yang terdiri dari bobot yang dapat dinormalisasi sesuai probabilitas. Adapun pemanfaatan LDA telah banyak digunakan dalam berbagai bidang, diantaranya untuk analisis trend pada media sosial [15], mendeteksi topik untuk pelacakan konten percakapan [24], dan telah terbukti mampu berkerja dengan baik untuk dokumen panjang seperti artikel Wikipedia maupun dokumen pendek seperti tweet [23].

2.2.6 Validasi Topik menggunakan Perplexity

Perplexity menjadi ukuran kualitas standar untuk model topik. Perplexity mengukur kemampuan model topik untuk menggeneralisasi dokumen setelah memperkirakan model menggunakan dokumen

latih. Perplexity yang lebih rendah berarti kemampuan generalisasi yang lebih baik [20]. Metode validasi perplexity merupakan sebuah metode yang digunakan untuk menguji ketepatan atau kesesuaian informasi dari dokumen dengan topik yang dihasilkan. *Perplexity* mengambil n sampel dari N populasi data untuk diuji, apakah n sampel tersebut memiliki kesesuaian topik dengan kelompok topik dalam N populasi[20].

Halaman ini sengaja dikosongkan

BAB 3

METODOLOGI

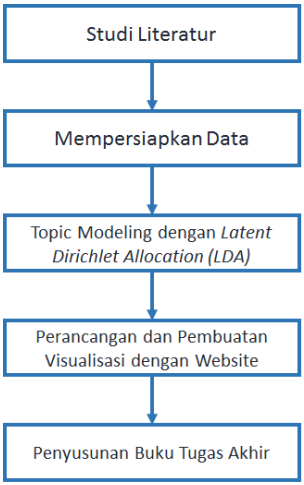
Pada bab metode penelitian akan dijelaskan mengenai tahapan – tahapan apa saja yang dilakukan dalam pengerjaan tugas akhir ini beserta deskripsi dan penjelasan tiap tahapan tersebut. Lalu disertakan jadwal pengerjaan tiap tahapanan.

3.1 Tahapan pengerjaan tugas akhir

Pada sub bab ini akan menjelaskan mengenai metodologi dalam pelaksanaan tugas akhir. Metodologi ini dapat dilihat pada Gambar 3.1

3.1.1 Studi literatur

Tahap studi literatur disini dilakukan dengan tujuan untuk dapat memahami konsep, metode, dan teknologi sesuai bahasan dan permasalahan sehingga dapat memberi solusi mengenai permasalahan yang akan digunakan dalam penyusunan tugas akhir. Adapun literatur yang digunakan dalam penelitian ini adalah terkait *Social Media Analysis* oleh Yuheng Hu, Lydia Manikonda, dan Subbarao Kambhampati [11] dan *Topic Modelling, LDA* mengacu pada penelitian David M. Blei [5].

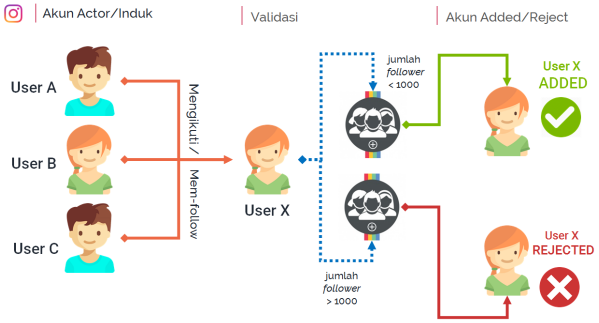


Gambar 3.1: Metodologi Penelitian

3.1.2 Mempersiapkan Data

Tahap mempersiapkan data terdiri dari beberapa aktifitas, yakni, pengumpulan data akun *Instagram*, pemilihan akun *Instagram*, penambahan dan validasi akun *Instagram*, pengumpulan *caption* akun *Instagram* terpilih untuk selanjutnya dilakukan pemrosesan data *caption*, rangkaian tahapan ini dilakukan untuk mempersiapkan dokumen yang akan dianalisis menggunakan topic modelling *LDA*. Adapun pengambilan data dokumen yang akan dianalisis adalah data media sosial milik akun siswa dari 18 sekolah SMP di Surabaya meliputi: (SMPN 1, SMPN 2, SMPN 30, SMPN 23, SMPN 35, SMPN 13, SMP Muhammadiyah 9, SMP Muhammadiyah 5, SMPN 12, SMP Muhammadiyah 5, SMPN 6, SMPN 19, SMPN 29, SMPN 04, SMPN 44, SMPN 15, SMPN 18, dan SMPN 45) dengan memanfaatkan crawler.

Gambar 3.2: Contoh proses pengumpulan dan pemilihan data akun*Instagram* salah satu sekolah SMP



Gambar 3.3: Contoh proses penambahan dan validasi akun *Instagram* siswa SMP

1. Pengumpulan dan pemilihan data akun Pada gambar 3.2 menjelaskan salah satu contoh proses pengumpulan data dimulai dengan pengambilan data akun berdasarkan hasil kuisioner mata kuliah Etika Profesi, kemudian dilakukan proses pemilihan untuk akun yang valid dan tidak valid serta dilakukan pengelompokan sesuai dengan sekolah masing-masing.
2. Penambahan dan validasi akun Pada gambar **??** menjelaskan proses penambahan akun untuk menggantikan akun yang tidak valid, penambahan data akun dilakukan dengan cara mengambil beberapa data *actor*/data induk untuk kemudian

Tabel 3.1: Tabel detail akun user

TOTAL AKUN	DETAIL AKUN	
	Jumlah Akun Private	**Jumlah Akun Public**
495	104	391
Presentase	21	79

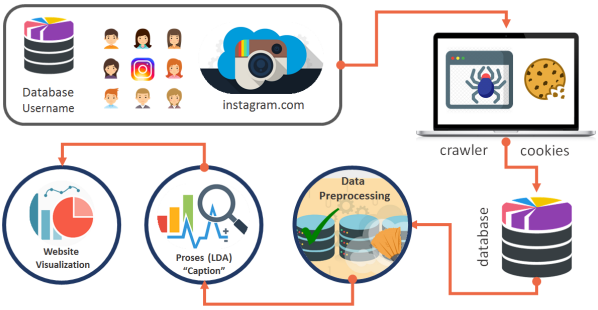
dilihat akun yang ia *follow*, dari akun-akun yang di *follow* oleh masing-masing akun induk, kemudian dicek apakah ada kesamaan antar akun yang di*follow*, apabila ada akun tertentu yang sama, kemudian dilihat *follower* akun tersebut apabila masih wajar, dalam arti tidak lebih dari 1000 maka akun tersebut dapat dikatakan valid akun siswa SMP dan captionnya menjadi *ADDED*, metode ini mengadopsi metode *tracking actor* [22], sehingga didapatkan data akun siswa SMP sebanyak 495 akun, dengan detail jumlah akun kategori *public* sebanyak 391 (79 persen) dan akun *private* sebanyak 104 (21 persen), seperti yang ditunjukkan tabel.

3. Pengumpulan caption dilakukan dengan melakukan crawling terhadap akun Instagram berdasarkan nama akun yang ada di dalam database kemudian data *caption* disimpan dalam database, secara skema seperti yang digambarkan pada gambar 3.4:

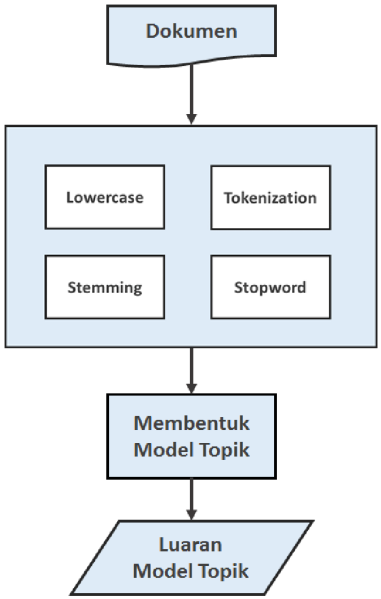
3.1.3 *Topic modelling* dengan *Latent Dirichlet Allocation*

Tahap *topic modelling* dengan *Latent Dirichlet Allocation* (LDA) terhadap dokumen (caption) Instagram dalam beberapa tahap digambarkan melalui gambar 3.5:

1. Pra-pemrosesan Corpus Dalam melakukan topic modelling dengan LDA, diperlukan langkah-langkah untuk untuk mem-



Gambar 3.4: Contoh proses pengumpulan data *caption* akun *Instagram* siswa SMP



Gambar 3.5: Tahap *topic modelling* dengan *Latent Dirichlet Allocation (LDA)* dan pra-pemrosesan *corpus*

Tabel 3.2: Contoh data sebelum dan sesudah melalui tahap *lowercase*

Before Lowercase	*After Lowercase*
Indahnya hidup jika kita menatap ke depan.. Dan Lengkapnya kehidupan jika di samping kita ada org yg rela meminjamkan pundakya disaat kita lelah untuk melaju kedepan	indahnya hidup jika kita menatap ke depan.. dan lengkapnya kehidupan jika di samping kita ada org yg rela meminjamkan pundakya disaat kita lelah untuk melaju kedepan

persiapkan data sehingga dapat diolah pada tahap berikutnya, adapun tahapan preprocessing ada beberapa, yang pertama adalah *lowercase* dimana data teks perlu dibentuk menjadi menjadi lowercase dengan tujuan agar kata yang sama namun berbeda secara penulisan huruf kapital dan tidak, tidak dianggap kata sebagai yang berbeda. Contoh data sebelum dan sesudah melalui tahap lowercase ditunjukkan dengan tabel 3.2: tahap selanjutnya adalah *tokenization*, dalam proses ini dilakukan pemisahan deretan kata di dalam kalimat atau paragraf menjadi potongan kata tunggal atau termmed word. Proses *tokenization* bertujuan untuk mempersiapkan dokumen untuk proses berikutnya, yaitu proses *stopwords* dan *stemming* agar dapat dilakukan. Contoh data sebelum dan sesudah melalui tahap *tokenization* ditunjukkan dengan tabel 3.3: Tahap berikutnya adalah *stopwords*, merupakan kata umum (common words) yang biasanya muncul dalam jumlah besar dan dianggap tidak memiliki makna. Contoh kelas kata termasuk stopwords ditunjukkan dengan tabel 3.4: Adapun daftar stopwords yang digunakan adalah stopwords Bahasa Indonesia yang disusun berdasarkan penelitian Fadillah Z Tala [22]. Beberapa contoh daftar stopwords yang

Tabel 3.3: Contoh data sebelum dan sesudah melalui tahap *tokenization*

Before Tokenization	*After Tokenization*
indahnya hidup jika	‘indahnya’ ‘hidup’ ‘jika’
kita menatap ke depan..	‘kita’ ‘menatap’ ‘ke’ ‘depan..’
dan lengkapnya kehidupan	‘dan’ ‘lengkapnya’ ‘kehidupan’
jika di samping kita ada	‘jika’ ‘di’ ‘samping’ ‘kita’ ‘ada’
org yg rela meminjamkan	‘org’ ‘yang’ ‘rela’ ‘meminjamkan’
pundakya disaat kita lelah	‘pundakya’,‘disaat’ ‘kita’ ‘lelah’
untuk melaju kedepan	‘untuk’,‘melaju’ ‘kedepan’

Tabel 3.4: Contoh kelas kata

Kelas Kata	**Contoh**
Kata Depan	di, pada, dari, ke, kepada, akan, oleh, daripada, hingga, sampai, dll
Kata Sambung	dan, dengan, serta atau, sedangkan, sebab, jika, bila, sebagai, sehingga, sesudah, dll
Kata Ganti	saya, aku, ku, kami, kita, kamu, engkau, anda, kalian, ia, dia, beliau, mereka, dll

Tabel 3.5: Contoh daftar stopwords berdasarkan penelitian Fadillah Z Tala [22]

Daftar Stopwords
‘ada’,‘agak’, ‘amatlah’, ‘bagaimana’, ‘bahwa’, ‘cuma’, ‘cukup’, ‘demikian’, ‘dapat’, ‘entah’, ‘guna’, ‘hal’, ‘hendak’, ‘ialah’, ‘ini’, ‘jika’, ‘juga’, ‘kira’,‘lalu’, ‘mana’, ‘memang’, ‘namun’, ‘oleh’, ‘pasti’, ‘para’, ‘saat’, ‘sangat’,‘tanpa’, ‘tiap’, ‘yakni’, ‘yaitu’

Tabel 3.6: Contoh data sebelum dan sesudah melalui tahap penghapusan *Stopwords*

Before Stopword Removal	***After Stopword Removal***
‘indahnya’ ‘hidup’ ‘jika’,‘kita’ ‘menatap’ ‘ke’ ‘depan..’ ‘dan’ ‘lengkapnya’ ‘kehidupan’ ‘jika’ ‘di’ ‘samping’ ‘kita’ ‘ada’ ‘org’ ‘yang’ ‘rela’ ‘meminjamkan’ ‘pundakya’,‘disaat’ ‘kita’ ‘lelah’ ‘untuk’,‘melaju’ ‘kedepan’	‘indahnya’ ‘hidup’ ‘menatap’ ‘lengkapnya’ ‘kehidupan’ ‘rela’ ‘meminjamkan’ ‘pundakya’,‘lelah’ ‘melaju’

tersimpan dalam daftar yang dimaksud tercantum pada tabel 3.5: Menghilangkan stopwords merupakan tahap yang penting, mengingat tingginya frekuensi kemunculan stopwords dalam dokumen, yang berujung pada tingginya probabilitas kata-kata stopwords dalam topik, sehingga topik tidak dapat diinterpretasi dengan baik. Contoh data sebelum dan sesudah melalui tahap penghapusan Stopwords ditampilkan pada 3.6: Tahap selanjutnya adalah Stemming, yang digunakan untuk mengganti bentuk dari suatu kata menjadi kata dasar dari kata tersebut. Stemming bekerja dengan menghilangkan semua imbuhan (affixes), baik yang terdiri dari awalan (pre-

Tabel 3.7: Contoh data sebelum dan sesudah melalui tahap *Stemming*

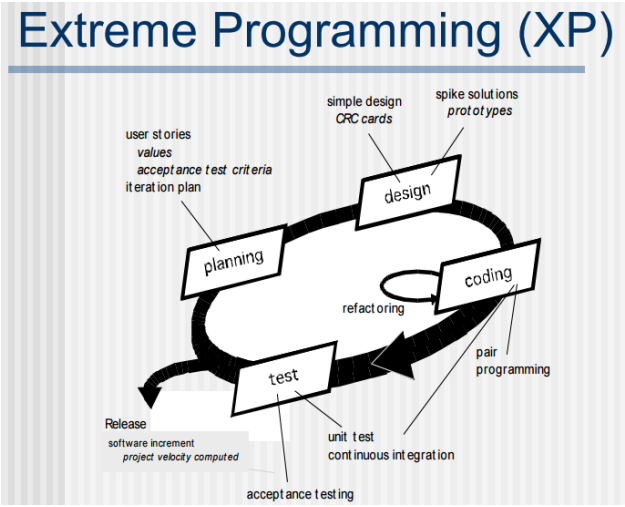
Before Stemming	***After Stemming***
‘indahnya’‘hidup’ ‘menatap’ ‘lengkapnya’ ‘kehidupan’ ‘rela’ ‘meminjamkan’ ‘pundakya’,‘lelah’ ‘melaju’	‘indah’ ‘hidup’ ‘tatap’ ‘lengkap’ ‘hidup’ ‘rela’ ‘pinjam’ ‘pundak’,‘lelah’ ‘laju’

fixes), sisipan (infixes), akhiran (suffixes) dan kombinasi dari awalan dan akhiran (confixes) pada kata turunan. Data teks perlu dibentuk menjadi menjadi kata dasarnya dengan tujuan agar tidak terdapat kata yang sama namun berbeda karena adanya imbuhan (affixes). Adapun proses stemming dilakukan dengan menggunakan library Sastrawi, yaitu library stemmer bahasa indonesia dengan lisensi MIT yang memanfaatkan kamus kata dasar dari Kateglo sebagai acuan. Contoh data sebelum dan sesudah melalui tahap Stemming ditampilkan pada tabel 3.7

2. Membentuk Model Topik Tahap membentuk topik model bertujuan untuk menghasilkan model topik yang paling tepat untuk dokumen. Model topik dikatakan tepat apabila mampu menghasilkan luaran yang baik pada tahap validasi model topik. Untuk menghasilkan model topik yang tepat, hal yang dilakukan adalah dengan melakukan eksperimen pada nilai input parameter. Adapun parameter yang dimaksud adalah number of topics dan words in topic. Parameter number of topics menentukan jumlah topik dalam satu dokumen, sementara parameter number of words in topic menunjukkan jumlah kata penyusun topic. Berikut merupakan contoh luaran model topik yang dilakukan dengan eksperimen *input parameter number of topics* sejumlah 5 dan *words in topic* sejumlah 6 yang ditunjukkan pada tabel 3.8

Tabel 3.8: Contoh luaran model topik hasil percobaan

Topic	Probabilitas*Kata
Captioned	0.020*sekolah + 0.017*teman + 0.007*belajar + 0.006*guru + 0.006*tugas + 0.006*susah
Selfie	0.009*kamera + 0.008*pemandangan + 0.007*cantik + 0.006*indah + 0.006*kenangan + 0.006*bahagia
Food	0.023*kenyang + 0.009*harga + 0.009*murah + 0.008*enak + 0.008*nikmat + 0.008*lezat
Fashion	0.033*keren + 0.022*baru + 0.012*sepatu + 0.011*tas + 0.011*bagus + 0.008*sayang
Gadget	0.033*hp + 0.028*pulsa + 0.027*sinyal + 0.024*lemot + 0.020*jam + 0.018*chat



Gambar 3.6: Alur proses *Extreme Programming* [18]

3.1.4 Perancangan dan Pembuatan Visualisasi dengan Aplikasi Web

Pada tahap ini dilakukan pengembangan aplikasi yang merupakan implementasi dari hasil melakukan analisa dan desain aplikasi. Aplikasi dikembangkan berdasarkan fungsionalitas yang telah didaftar dan arsitektur pada sistem yang telah didesain. Tahapan pengembangan menggunakan metode pengembangan *Agile Software Development*. Salah satu model yang digunakan dari *Agile Software Development* adalah *Extreme Programming*. Dimana model tersebut merupakan model yang paling sering digunakan dan menggunakan pendekatan *object-oriented* paradigma pengembangannya. Pada *Extreme Programming*, terdapat 4 proses utama yaitu *planning* dimana perencanaan dilakukan dengan memperhatikan kecepatan penyelesaian, selanjutnya adalah *design* dimana menggunakan prinsip KIS (keep it simple), *coding* dimana proses pengem-

bangan perangkat lunak dimulai, dan yang terakhir adalah *testing* dimana semua unit test diuji dan dievaluasi.[3] .

Pada proses Perencanaan dan Pengembangan aplikasi, pada penelitian ini dilakukan beberapa tahap yaitu:

1. Perencanaan
   Di tahap perencanaan, dilakukan proses penerjemahan kebutuhan aplikasi yang akan dikembangkan nantinya. Pada tahap ini semua fitur-fitur yang diharapkan pada aplikasi didokumentasikan. Setelah semua kriteria kebutuhan didefinisikan, selanjutnya adalah menentukan *timeline* pengerjaan hingga aplikasi siap diuji dan dipresentasikan.
2. Desain
   Pada tahap ini, hasil perencanaan pada tahap sebelumnya akan digunakan untuk membuat gambaran desain dari fitur dan fungsionalitas terhadap aplikasi yang akan dibangun nantinya. Pada tahap ini akan dibuat beberapa desain antara lain:
   - Desain database
   - Desain Crawler
   - Desain sistem
   - Desain User Interface
   - Prototype
3. Pengkodean
   Pada tahap ini, dilakukan pengkodean terhadap aplikasi monitoring yang meliputi :
   - Pengkodean *Web-Crawler*
     Disini *Web-Crawler* mulai dilakukan kodifikasi dimana *Web-Crawler* ini nanti berfungsi sebagai pengakuisisi data yang akan digunakan untuk melakukan pengambilan data, berupa *caption* dari posting foto siswa SMP, yang kemudian disimpan untuk dilakukan proses *topic modelling*.
   - Pengkodean Topic Modelling

Disini dilakukan pengkodean untuk melakukan preprocessing semua data caption yang sudah disimpan untuk selajutnya dilakukan *Topic Modelling* menggunakan metode LDA dengan framework Gensim dan Bahasa Python.

- Pengkodean Web Visualisasi
  Data yang sudah melewati proses *topic modelling* kemudian divisualkan ke dalam website dengan menunjukkan dashboard yang menggunakan bahasa PHP.

4. Pengujian Pada tahap ini dilakukan pengujian aplikasi untuk memastikan aplikasi dapat berjalan sesuai dengan yang diharapkan oleh pengguna dan mencatat semua *bug* dan *error* yang ada pada aplikasi. Pengujian pada tahap ini difokuskan pada:
   - Menguji ketepatan atau kesesuaian informasi dari dokumen dengan topik yang dihasilkan. Pada tahap ini dilakukan validasi terhadap hasil pengolahan data dengan LDA yang dilakukan untuk membuktikan bahwa distribusi topik yang dihasilkan memiliki kesesuaian dengan dokumen *caption instagram* siswa SMP. Validasi dilakukan dengan menggunakan *perplexity*.

Halaman ini sengaja dikosongkan

BAB 4

PERANCANGAN

Pada bab ini membahas terkait alur perancangan terkait beberapa hal yang diperlukan dalam proses pembuatan aplikasi sesuai dengan alur yang dijelaskan pada bab metodologi. Adapun perancangan ini diperlukan sebagai panduan dalam melakukan penelitian tugas akhir, yang dijelaskan sebagai berikut.

4.1 Pengambilan Data

Dalam pelaksanaan analisis dan visualisasi *topic modelling caption* instagram siswa SMP, data berupa caption merupakan objek utama yang analisis. Data yang dibutuhkan berupa data tipe teks hasil *crawling* media sosial instagram. Dalam melakukan pengumpulan data dibutuhkan *username* akun siswa SMP sebagai acuan lingkup data *caption* yang nantinya diambil. Adapun atribut pada tabel *username* ditunjukkan pada tabel 4.1 Proses penyimpanan data *caption* berdasarkan *username* akun, akan disimpan di *database caption* dimana attribut pada *database caption* dijelaskan melalui Tabel 4.2.

4.2 Metodologi Implementasi

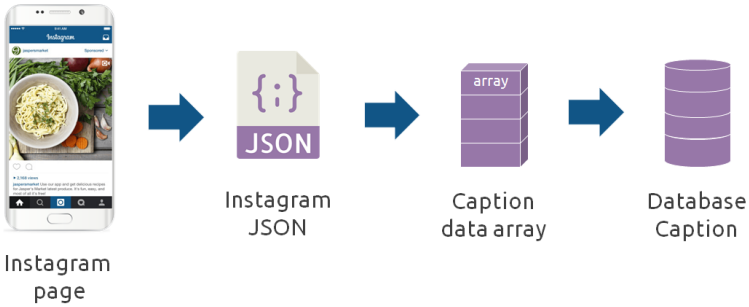
Metodologi implementasi penelitian merupakan tahapan dalam mencapai tujuan penelitian yang disesuaikan dengan komputasi secara otomatis. Komputasi yang dilakukan pada penelitian ini adalah dengan beberapa tahapan, yakni pemrosesan data, pencarian model

Tabel 4.1: Keterangan Attribut Database Username

Nama Attribut	Tipe Data	Keterangan
username	varchar (255)	Merupakan data username instagram siswa SMP
link	varchar (200)	Merupakan link halaman instagram milik siswa SMP
kode-sekolah	varchar (30)	Merupakan kode unik untuk masing-masing sekolah
gender	varchar (5)	Merupakan data gender akun siswa SMP

Tabel 4.2: Keterangan Attribut Database Caption

Nama Attribut	Tipe Data	Keterangan
Id	Varchar(255)	id untuk tiap posting milik user
Waktu	Int	Merupakan data waktu yang menunjukkan kapan user melakukan posting, data waktu yang ada pada instagram menggunakan format Unix epoch time
Oleh	Varchar(255)	Merupakan data berupa nama user siswa SMP yang melakukan aktivitas menggunakan akun instagramnya.
Pesan	Varchar(255)	Merupakan data berupa caption atau pesan yang ada pada tiap foto yang diunggah ke media sosial instagram
Foto	Varchar(255)	Merupakan data berupa link foto yang diunggah user ke media sosial instagram.



Gambar 4.1: Alur proses *crawling data*

serta klasifikasi menggunakan python adapun proses visualisasi hasil pengolahan data akan dilakukan menggunakan PHP. Dalam penelitian ini ada 6 tahapan utama dalam tahap melakukan implementasi yaitu *crawling data, load data*, pra-proses data, pemrosesan data, klasifikasi data, dan visualisasi data.

4.2.1 *Crawling Data*

Crawling data merupakan tahapan pengambilan data *caption instagram* yang kemudian akan digunakan dalam proses *topic modelling*. Tahap *crawling data* ini memanfaatkan nama akun setiap *user* yang sudah dicantumkan dalam database, tahap *crawling* ini menggunakan *PHP* dengan memanfaatkan *JSON* pada setiap halaman user instagram berbeda yang berisi informasi detail *posting* gambar untuk kemudian dimasukkan dalam bentuk *array* dan selanjutnya disimpan dalam database. Jumlah akun instagram yang digunakan adalah 495 akun aktif, dan jumlah *caption* yang berhasil didapatkan ada sebanyak 4664 data untuk periode Januari hingga Juni 2017. Gambaran proses *crawling* dijelaskan pada gambar 4.1:

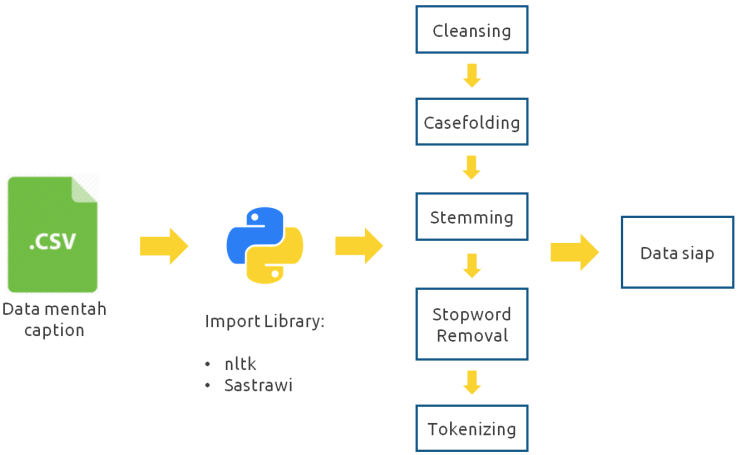
4.2.2 *Load Data*

Load data merupakan tahapan dimana data diproses untuk dapat dibaca ke dalam tools sebelum melakukan analisa pada penelitian. Data asli atau *raw data* yang berhasil diambil kemudian diunduh ke dalam format *.csv*. Menggunakan pyhton, *load data* dengan format csv dapat dilakukan dengan menggunakan modul csv. Adapun data yang dimuat adalah data *caption instagram* siswa SMP dari bulan Januari 2017 hingga bulan April 2017.

4.2.3 Pra-proses Data

Tahap pra-proses data mencakup beberapa langkah utama pengerjaan yakni pembersihan data dari *tag* atau karakter-karakter tertentu, pengubahan data menjadi huruf kecil *casefolding*, stemming, stopword removal, serta tokenization. Untuk penjelasan secara lebih detail, dijelaskan pada Gambar 4.2:

1. *Casefolding* dilakukan untuk merubah struktur kata menjadi huruf kecil.
2. *Cleansing* adalah tahap untuk menghilangkan karakter-karakter yang tidak terpakai pada data *caption*
3. *Stemming* dilakukan untuk menghilangkan kata imbuhan serta mengubah kata-kata pada data menjadi kata dasar yang memanfaatkan modul Sastrawi.
4. *Stopword* dilakukan dengan mendaftar kata-kata dalam bahasa indonesia, merujuk pada penelitian Fadillah Z Tala yaitu dengan memasukkan kata yang sering digunakan namun tidak memiliki nilai informasi, pada studi kasus penelitian ini dibuat stopword khusus untuk gaya bahasa siswa usia remaja yang mayoritas menggunakan bahasa kekinian yang berkembang di era-nya.



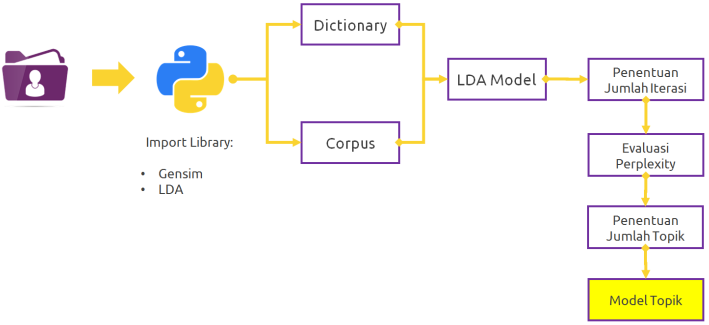
Gambar 4.2: Alur pra-proses data

5. Tokenization dilakukan untuk memecah atau memotong *string* pada kalimat menjadi tiap kata yang menyusunnya, dalam proses tokenizing ini digunakan modul *nltk*.

Setelah data melewati pra-proses, kemudian data diterjemahkan untuk menjadi corpus serta disimpan untuk penggunaan pemrosesan data selanjutnya.

4.2.4 Proses Data

Pada tahapan proses data, langkah utama yang dilakukan adalah melakukan pencarian model dengan menggunakan modul LDA. Pada Gambar 4.3 menjelaskan tentang bagaimana alur pemrosesan data dijalankan.



Gambar 4.3: Alur Pemrosesan Data

1. Pembentukan *Dictionary* dan *Corpus*
   Dalam melakukan *topic modelling*, data terlebih dahulu perlu dirubah ke dalam bentuk *dictionary* dan *corpus*. Pengertian *Dictionary* adalah format data yang mengandung himpunan kata unik yang memiliki indeks, sehingga dapat memudahkan dalam menampilkan kata yang termasuk dalam sebuah model, sedangkan *Corpus* merupakan format data yang memiliki bentuk dokumen *term-matrix*, yang nantinya berguna guna melakukan eksperimen pembentukan model. Adapun proses pembentukan *dictionary* dan *corpus* ditunjukkan dengan gambar 4.4
2. *Topic modelling* dengan LDA
   Pada tahap *topic modelling* menggunakan LDA, langkah yang perlu dilakukan adalah membentuk model dengan menggunakan *library gensim*, kemudian model dievaluasi menggunakan *perplexity*. Dalam proses membentuk model percobaan *input parameter* sangat dibutuhkan. Hasil dari pencarian model akan digunakan untuk mendapatkan topik apa yang muncul dari analisis pada dokumen. Setelah model topik di-

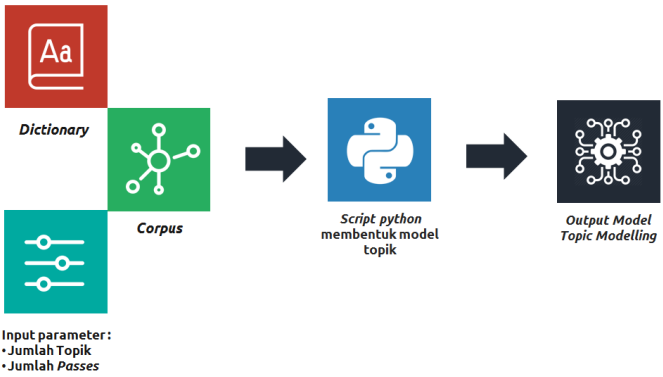
Gambar 4.4: Alur pembentukan *dictionary* dan *corpus*

dapatkan kemudian, dilakukan evaluasi *perplexity* menggunakan modul *logging*. Model dapat dikatakan terbaik apabila menunjukkan hasil *perplexity* yang lebih kecil dan stabil. Adapun skema proses *topic modelling* ditunjukkan dengan gambar 4.5

4.2.5 Validasi Topik Model

Tahap validasi topik memiliki tujuan untuk memastikan model topik yang dihasilkan dari hasil topic modelling pada dokumen adalah sesuai, baik topik maupun kata-kata yang terkandung dalam topik tersebut. Adapun beberapa hal yang dianalisis dalam tahap validasi topik model adalah

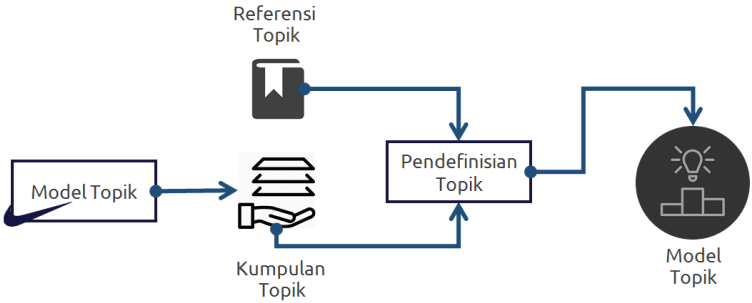
1. Jumlah iterasi yang tepat untuk membentuk topik model.
2. Jumlah topik yang sesuai berdasarkan distribusi perplexity.



Gambar 4.5: Alur topic modeling dengan LDA

4.2.6 Analisa Topik

Analisis topik digunakan untuk mengidentifikasi hasil topik yang diterjemahkan dari proses LDA terhadap jumlah topik yang dipilih. Gambar 5.21 menampilkan alur dari analisis topik dalam penelitian ini. Analisis topik dilakukan dengan cara menerjemahkan topik-topik yang muncul dari setiap kata-kata disesuaikan dengan dokumen yang ada. Topik diterjemahkan kemudian dianalisis terhadap kesamaan satu topik dengan topik yang lainnya. Kemudian proses pendefinisian topik dilakukan dengan cara mengacu pada jurnal referensi topik instagram secara global, selanjutnya akan terbentuk topik-topik yang dianggap sesuai dengan studi kasus dalam penelitian ini.



Gambar 4.6: Alur Analisis Topik

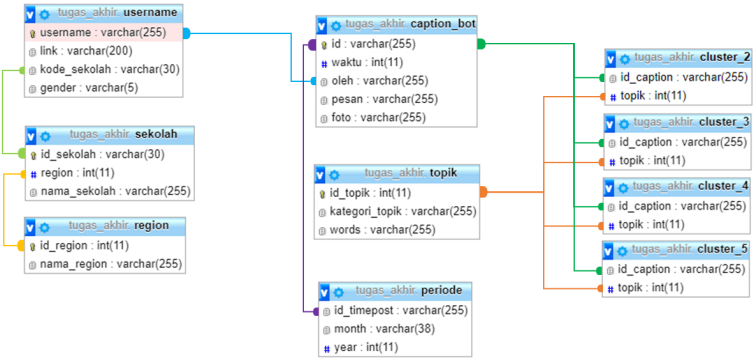
4.2.7 Klasifikasi Data

Klasifikasi data merupakan tahap yang digunakan dalam menentukan dokumen yang ada pada data caption untuk dicocokkan dengan topik yang telah diidentifikasi. Klasifikasi data dilakukan dengan menggunakan probabilitas terhadap kesesuaian dokumen dengan topik yang dihasilkan.

4.2.8 Konstruksi Perangkat Lunak

Berikut ini adalah perancangan perangkat lunak untuk visualisasi dashboard Aplikasi *TeenStagram*, meliputi perancangan database, dan antarmuka aplikasi.

1. Design Database Dalam penyusunan perangkat lunak *TeenStagram* digunakan *database schema* yang dijelaskan pada gambar 4.7:
   Pada schema tersebut terdapat entitas:
   - ***caption bot*** yang menampung data yang berhubungan dengan informasi terkait *caption* siswa SMP, yakni *id*



Gambar 4.7: *database schema* yang digunakan dalam pegerjaan tugas akhir

caption, waktu, nama pengguna, pesan dan tautan foto akun instagram siswa SMP yang kemudian akan diolah untuk proses topic modeling.

- ***username*** yang menampung data *username* yang mendukung proses *crawling caption*.
- **sekolah** yang menampung kode dan nama sekolah.
- **region** yang menampung kode dan nama wilayah dimana sekolah tertentu berada.
- **topik** yang berisi *id topik*, pendefinisian nama kategori topik dan kata-kata yang menyusun suatu topik tertentu.
- ***Cluster 2*** yang mengandung id caption dan pendefinisian id topik hasil pemrosesan menggunakan python sesuai dengan model dua topik
- ***Cluster 3*** yang mengandung id caption dan pendefinisian id topik hasil pemrosesan menggunakan python sesuai dengan model tiga topik
- ***Cluster 4*** yang mengandung id caption dan pendefinisian id topik hasil pemrosesan menggunakan python se-

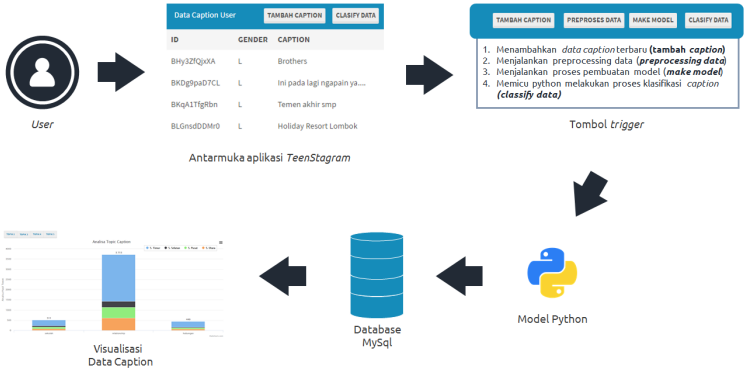
suai dengan model empat topik
    - ***Cluster 5*** yang mengandung id caption dan pendefinisian id topik hasil pemrosesan menggunakan python sesuai dengan model lima topik
- ***Periode*** yang mengandung $id_timepost, month, danyearyangdigunakans$

4.2.9 Integrasi antara PHP dan Python

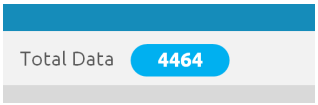
Pada tahapan ini dilakukan proses integrasi data melalui proses penambahan data akun dengan proses crawling, pembuatan model cluster dan klasifikasi data caption menggunakan pengkodean PHP dengan menjalankan fungsi *Python* yang sebelumnya sudah dibuat, selanjutnya data akan disimpan ke dalam database, untuk kemudian divisualkan menggunakan PHP. Penjelasan secara detail dari skema integrasi kode PHP dan model python ditunjukkan pada Gambar 5.24 User dapat melakukan tambah data melalui antarmuka aplikasi, kemudian ketika user menekan tombol classify data, merupakan trigger untuk menjalankan model python untuk klasifikasi data, hasil dari proses tersebut kemudian disimpan ke dalam database dan kemudian divisualisasikan menggunakan kode PHP dengan bar chart.

4.2.10 Desain Antarmuka Aplikasi Visualisasi

Melalui model yamg sudah ada didapatkan hasil klasifikasi data yang siap divisualkan ke dalam beberapa kategori sesuai dengan proses topic modeling. Visualisasi yang digunakan dalam penelitian ini adalah menggunakan bahasa pemrograman PHP, dimana user dapat melihatnya ke dalam sebuah dashboard. Rancangan dashboard yang akan ditampilkan dari analisa topic modelling, dibedakan menjadi beberapa konten, yakni, berdasarkan gender, wilayah sekolah, dan periode waktu. Pada Gambar 5.2 menjelaskan tentang jum-

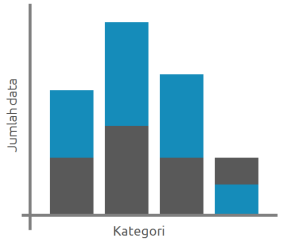


Gambar 4.8: Alur proses integrasi antarmuka aplikasi dengan model

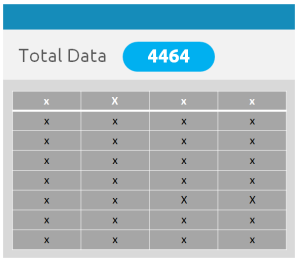


Gambar 4.9: Alur proses integrasi antarmuka aplikasi dengan model

lah data caption yang berhasil terakuisisi, sedangkan Gambar 4.10 menjelaskan tentang diagram topic yang terbentuk dari data caption, dan Gambar 4.11 menampilkan rancangan tampilan tabel untuk menunjukkan data berupa ***id caption, jenis kelamin akun siswa, bulan, tahun, kategori topik dan pesan/caption***, yang ditampilkan ke dalam aplikasi TeenStagram.



Gambar 4.10: Distribusi topik berdasarkan kategori



Gambar 4.11: *Data caption* yang berhasil terakuisisi

Halaman ini sengaja dikosongkan

BAB 5

IMPLEMENTASI

Bab ini menjelaskan hasil dari implementasi perancangan studi kasus atau hasil dari proses pelaksanaan penelitian. Hasil yang akan dijabarkan adalah hasil eksperimen terhadap data yang digunakan sebagai acuan penelitian. Selain itu, akan dijelaskan juga mengenai tantangan dan kesulitfan dalam proses pelaksanaan penelitian.

5.1 Lingkungan Implementasi

Dalam pelaksanaan identifikasi dan visualisasi *topic caption* akun *Instagram* siswa SMP di Surabaya, dibutuhkan perangkat-perangkat untuk menunjang keberlangsungan penelitian. Adapun perangkat-perangkat yang dibutuhkan berupa perangkat keras dan spesifikasinya ditunjukkan dengan Tabel 5.1. Kemudian untuk perangkat lunak yang digunakan dalam implementasi model ditunjukkan dalam tabel 5.2. Selain menggunakan *hardware* dan *software*, dalam penelitian ini juga digunakan beberapa *library* untuk mendukung proses *topic modeling* menggunakan *python*, adapun beberapa *library* yang digunakan ditunjukkan dalam tabel 5.3.

5.2 Pengambilan Data

Pengambilan data adalah tahap awal dimana data didapatkan dari hasil crawling halaman akun user Instagram yang namanya telah tercantum dalam database. Tahap ini dilakukan melalui pengumpulan data akun siswa terlebih dahulu.

Tabel 5.1: Spesifikasi Perangkat Keras

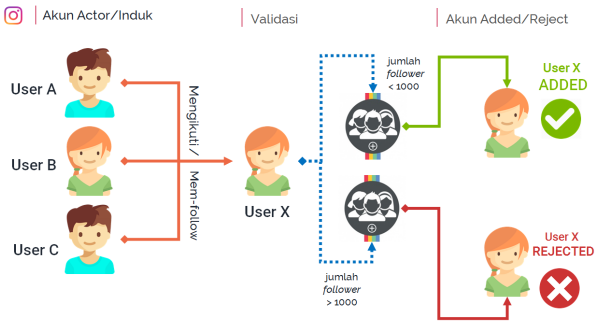
No.	Hardware	Spesifikasi
1.	Jenis	Samsung NSeries 535U4C
2.	Processor	AMD A8-4555M APU with Radeon (tm) HD Graphics(4CPUs),∼1.6GHz
3.	RAM	6144 MB RAM
4.	Hardisk	700 GB

Tabel 5.2: Spesifikasi Perangkat Lunak

No.	Software	Penggunaan
1.	Windows10 Pro 64-bit	Sistem Operasi
2.	Xampp,5.6.15 & PHP5.6.15	Webserver
3.	Python 3.5 64-bit (IDE Pycharm 2017.1)	Pemrosesan Data
4.	DBMS MySQL	Database Penyimpanan
5.	Ms.Excel 2016	Mengolah Angka
6.	Regexer	Pengolah Teks

Tabel 5.3: Daftar*Library yang digunakan*

No.	Library	Penggunaan
1.	nltk (3.2.2)	Preprocess Data
2.	Sastrawi (1.0.1)	Stemming
3.	gensim (2.0.0)	Topic modeling
4.	Logging	Perplexity Evaluation



Gambar 5.1: Contoh proses penambahan dan validasi akun *Instagram* siswa SMP

Pengumpulan Data Akun

Proses pengumpulan data akun *instagram* didapatkan dari hasil rekap kuisioner sosialisasi etika profesi, kemudian dilakukan proses pemilihan untuk akun yang valid dan tidak valid serta dilakukan pengelompokan sesuai dengan sekolah masing-masing.

- Penambahan dan validasi akun

Pada gambar 5.1 menjelaskan proses penambahan akun untuk menggantikan akun yang tidak valid, penambahan data akun dilakukan dengan cara mengambil beberapa data *actor*/data induk untuk kemudian dilihat akun yang ia *follow*, dari akun-akun yang di *follow* oleh masing-masing akun induk, kemudian dicek apakah ada kesamaan antar akun yang di*follow*, apabila ada akun tertentu yang sama, kemudian dilihat *follower* akun tersebut apabila masih wajar, dalam arti tidak lebih dari 1000 maka akun tersebut dapat dikatakan valid akun siswa SMP dan captionnya menjadi *ADDED*, metode ini mengadopsi metode *tracking actor* [22], sehingga didapatkan data akun siswa SMP sebanyak 495 akun, dengan detail jumlah akun

Tabel 5.4: Tabel detail akun user

TOTAL AKUN	DETAIL AKUN	
	Jumlah Akun Private	**Jumlah Akun Public**
495	104	391
Presentase	21	79

kategori *public* sebanyak 391 (79 persen) dan akun *private* sebanyak 104 (21 persen), seperti yang ditunjukkan tabel. Setelah data akun valid tersimpan dalam database, kemudian dilakukan proses *crawling caption* menggunakan pengkodean *PHP*.

Crawling Data

Proses *crawling data caption* dilakukan menggunakan *library CURL*, dan menggunakan *API* yang berguna untuk menguraikan *JSON* ke dalam *array*, adapun Kode 5.1 menunjukkan tentang pembuatan API yang digunakan untuk proses *crawling*

```
        if ( $_csc->uri[2] == 'instasearch' ) {
            $_url = 'https://www.instagram.com';

        if ( ! $_csc->uri[4] ) {
                $c->bc = $c->get( $_url . '/' . $_csc->uri[3]
                    . '/' );
                $x = 1;
                $json = json_decode( $c->xp( '<script type="
                    text/javascript">window._sharedData =', '
                    ;</script>' ) );
                foreach ( $json->entry_data->ProfilePage[0]->
                    user->media->nodes as $r ) {
                    $y = $x-1;
                    $Arr['photo'][$y]['code'] = $r->code;
                    $Arr['photo'][$y]['waktu'] = $r->date
                        ;
                    $Arr['photo'][$y]['caption'] = $r->
                        caption;
                    $Arr['photo'][$y]['display_src'] = $r
```

```
                    ->display_src;
                $x++;
            }
        .
```

Kode 5.1: Potongan *script* pembuatan *API* untuk proses *crawling data*

Setelah pembuatan *API* kemudian yang perlu diperhatikan adalah sistem keamanan *instagram* yang mampu mencurigai sebuah akun tergolong sebuah robot, oleh sebab itu untuk mengatasinya perlu dibuat sebuah *fake account* yang aktivitasnya dibuat semirip mungkin dengan akun pada umumnya, seperti, melakukan update foto, mengikuti akun orang lain, dan lain-lain. Pada penelitian ini dibuat sebuah akun bernama "Putriayuxxx" yang memiliki persona sebagai siswi usia SMP. Setelah fake account sudah terbentuk dan berprilaku seperti akun pada umumnya kemudian yang perlu dilakukan adalah memanfaatkan *cookies* milik akun "Putriayuxxx" untuk mempermudah proses *crawling data*, dengan maksud agar *instagram* tidak mencurigai akun yang digunakan tersebut sebagai robot. Adapun pengkodean untuk memanfaatkan *cookies* tersebut ditunjukkan dengan Kode 5.3:

```
        $username = "putriayuxxx";
        $password = "juara!!!";
        $useragent = "Mozilla/5.0 (X11; Linux x86_64)
            AppleWebKit/537.36 (KHTML, like Gecko) Ubuntu
            Chromium/50.0.2661.102 Chrome/50.0.2661.102
            Safari/537.36";
        $cookie=$username.".txt";

        @unlink(dirname(__FILE__)."/".$cookie);

        $url="https://www.instagram.com/accounts/login/?
            force_classic_login";

        $ch  = curl_init();

        $arrSetHeaders = array(
            "User-Agent: $useragent",
```

```
            'Accept: text/html, application/xhtml+xml,
                application/xml;q=0.9,*/*;q=0.8',
            'Accept-Language: en-US,en;q=0.5',
            'Accept-Encoding: deflate, br',
            'Connection: keep-alive',
            'cache-control: max-age=0',
        );
```

Kode 5.2: Potongan *cookies* pemanfaatan *cookies user*

Tahap berikutnya setelah pembuatan *API* dan pemanfaatan *cookies*, dilakukan pembuatan *script* untuk melakukan *crawling* berdasarkan *username* yang telah tersimpan sebelumnya. Adapun Kode 5.3 menunjukkan fungsi untuk *crawling data caption*.

```
    function input ( $i, $username ) {
        global $db;
        $f = $db->f( 'caption_bot', 'id', 'WHERE id=?', $i->
            code );
        if ( ! $f && $i->caption) {
                // var_dump( $t);
                $arr = array( $i->code, $i->waktu, $username,
                     removeEmoji($i->caption), $i->
                    display_src);
                var_dump($arr);
                        $db->i( 'caption_bot', 'id, waktu,
                            oleh, pesan, foto', $arr );
                } else return true;
        }
        function ulang ( $i1, $username, $n=0 ) {
                $j = apiJson( "http://kp.intip.in/api/
                    instasearch/$username/?np=$i1" );
                if (is_array($j->photo)) foreach ( $j->photo
                    as $t ) {
                        $habis = masukkin( $t, $username );
                        if ( $habis ) break;
                }
                if ( ! $habis && $j->next ) return ulangin(
                    $j->beda, $username, $n+1 );
                else return $n;
        }
        $db->pfx="";
        $f = $db->r( 'username', '*' );

        foreach ( $f as $r ) {
```

```
            $username = $r['username'];
            //$username = 'tethavaliant';
            $j = apiJson( 'http://kp.intip.in/api/
                instasearch/' . $username );
            echo "data terambil". '<br>';
            if (is_array($j->photo)) foreach ( $j->photo
                as $t ) {
                    $habis = masukkin( $t, $username );
                    if ( $habis ) break;
            }

            if ( ! $habis ) $n = ulangin( $j->beda,
                $username );
            echo 'Done repeating ' . $n . ' times.';
    }
```

Kode 5.3: Potongan *script* fungsi *crawling* berdasarkan *akun username instagram*

5.3 Memuat Data

Data *caption* yang telah didapatkan dari hasil *crawling* dan tersimpan di dalam database kemudian akan dimuat ke dalam bentuk *csv* untuk dan diolah ke tahap selanjutnya. Langkah pengkodean yang dilakukan untuk memuat data adalah dengan menggunakan *library csv*, Kode 5.4 menunjukkan proses memuat data menggunakan *library csv*

```
    import csv
    with open('data/1.data_training.csv') as raw:
        reader = csv.reader(raw)
        for row in reader:
            joim = ''.join(row)
```

Kode 5.4: Potongan *script* untuk memuat data

5.4 Pra-Proses Data

Dalam melakukan analisa model topik *caption instagram* siswa SMP di Surabaya, pra-proses data merupakan salah satu tahapan yang penting dalam penelitian, agar data dapat diolah ke proses berikutnya. Pra-proses data meliputi beberapa tahapan, yaitu ***case folding***, ***stemming***, ***stopword removal***, dan ***tokenization***.

5.4.1 *Case Folding*

Tahapan *case folding* adalah proses dimana data diubah ke dalam bentuk huruf kecil, dengan tujuan untuk menyamaratakan format. Adapun pengkodean yang digunakan untuk melakukan *case folding* adalah dengan menggunakan Kode 5.5.

```
            .lower()
```

Kode 5.5: *Method* untuk melakukan *case folding*

5.4.2 *Stemming*

Stemming merupakan tahapan untuk mengubah sebuah kalimat menjadi kata dasar. Dalam penelitian ini, stemming dilakukan dengan menggunakan *library Sastrawi* dengan menggunakan fungsi *stem*. Adapun Untuk melakukan proses *stemming* data, dapat dilakukan dengan menggunakan Kode 5.6.

```
            from nltk.tokenize import RegexpTokenizer
            from Sastrawi.Stemmer.StemmerFactory import
                StemmerFactory
            # create stemmer
            factory = StemmerFactory()
            stemmer = factory.create_stemmer()
            with open('data/1.data_training.csv') as raw:
```

```
reader = csv.reader(raw)
for row in reader:
    join = ''.join(row ).lower()
    token = tokenizer.tokenize(stemmer.stem(
        join))
```

Kode 5.6: Potongan *script* untuk melakukan *stemming* menggunakan *library* Sastrawi

5.4.3 Pendefinisian Stopword

Stopword removal merupakan tahap yang digunakan untuk menghilangkan kata-kata yang tidak memiliki makna dalam sebuah kalimat. Menurut Tim Depdikbud RI yang dipelopori oleh Hasan Alwi dkk. [1] *Stopword* didefinisikan ke dalam kategori kelas kata. Adapun beberapa kategori kelas kata yang tergolong ke dalam *stopword* adalah **kata keterangan, kata ganti, kata bilangan, kata depan, kata sambung, makna kata sambung, kata seru, kata sandang** dan**partikel penegas**.
Proses pembuatan *stopword* untuk penelitian ini sedikit unik karena studi kasus penelitian menyangkut caption instagram remaja yang notabene memiliki kosakata baru dan luas. Oleh karena itu proses pembuatan stopword dibuat melalui proses penghitungan kata yang sering muncul. Kemudian dari kata yang sering muncul tersebut, akan didefinisikan apakah kata tersebut memiliki makna atau tidak, sesuai dengan justifikasi yang mengacu pada konteks kata tugas [1]. Apabila suatu kata dinilai tidak memiliki makna, maka kata tersebut akan masuk ke dalam daftar kata yang dibuang *stopword*. Adapun pengkodean yang digunakan untuk mengetahui kata yang sering muncul adalah Kode 5.7

```
import csv
word_counter = {}
hasil=[]
```

```
with open('data/2.data_training_no_stem.csv') as
    raw:
    reader = csv.reader(raw)
    for row in reader:
        for kata in row:
        # asli= ' '.join(row)
            hasil.append(kata)
# print(hasil)
for word in hasil:
    if word in word_counter:
            word_counter[word] += 1
    else:
            word_counter[word] = 1

popular_words = sorted(word_counter, key =
    word_counter.get, reverse = True)

top_500 = popular_words[:500]
print(500)

with open('3.kata_dibuang.csv', 'w', newline='')
    as tulisFile:
    tulisFileWriter = csv.writer(tulisFile)
    for row in top_500:
        tulisFileWriter.writerow([row])
tulisFile.close()
```

Kode 5.7: Potongan *script* untuk menghitung kata yang sering muncul

Semua daftar kata *stopword* kemudian disimpan ke dalam sebuah *file.txt*, dan selanjutnya digunakan untuk proses *stopword removal*. Kode 5.8 menunjukkan pengkodean untuk melakukan proses *stopword removal*

```
with open('data/bikin_stopword1.txt') as f:
content = f.readlines()
# you may also want to .txt'
content = [x.strip() for x in content]
# create English stop words list
indo_stop = content
```

Kode 5.8: Potongan *script* untuk melakukan *stopword removal*

5.4.4 Tokenization

Tokenization merupakan tahap memecah teks yang dapat berupa kalimat, paragraf atau dokumen, menjadi token-token atau sekumpulan kata. *Tokenization* memisahkan teks berdasarkan spasi dan dijadikan kata dalam korpus. Untuk melakukan proses *tokenization*, dalam penelitian ini menggunakan *library NLTK* ***Natural Language Toolkit*** meliputi fungsi *tokenization dan regular expresion*. Berikut adalah pengkodean untuk melakukan *tokenization* menggunakan NLTK:

```
from nltk.tokenize import RegexpTokenizer
tokenizer = RegexpTokenizer(r'\w+')
with open('data/1.data_training.csv') as raw:
    reader = csv.reader(raw)
    for row in reader:
        join = ''.join(row ).lower()
        token = tokenizer.tokenize(stemmer.stem(
            join))
```

Kode 5.9: Potongan *script* untuk melakukan *tokenization*

Setelah melalui pra-proses data, hasil proses tersebut akan disimpan menjadi *file csv* untuk kemudian melalui tahap proses berikutnya, secara keseluruhan pengkodean Pra-proses data dapat dilihat pada Kode 5.10:

```
import csv

from nltk.tokenize import RegexpTokenizer
from Sastrawi.Stemmer.StemmerFactory import
    StemmerFactory
# create stemmer
factory = StemmerFactory()
stemmer = factory.create_stemmer()
with open('data/bikin_stopword1.txt') as f:
    content = f.readlines()
# you may also want to .txt'
content = [x.strip() for x in content]

# create English stop words list
```

```
indo_stop = content
tokenizer = RegexpTokenizer(r'\w+')
i=0
hasil=[]
with open('data/1.data_training.csv') as raw:
    reader = csv.reader(raw)
    for row in reader:
        join = ''.join(row ).lower()
        token = tokenizer.tokenize(stemmer.stem(
            join))
        hasil.append([i for i in token if not i
            in indo_stop])
        print(i)
        i=i+1

raw.close()

with open('data/2.data_training_with_stem.csv', '
    w',newline='') as tulisFile:
    tulisFileWriter = csv.writer(tulisFile)
    for row in hasil:
        tulisFileWriter.writerow([row])

    tulisFile.close()          token = tokenizer.
        tokenize(stemmer.stem(join))
```

Kode 5.10: Pengkodean Pra-pemrosesan data

5.5 Proses Data

Tahapan dalam pemrosesan data diawali dengan melakukan penyimpanan *dictionary* yang didapatkan dari tahap pra-proses data sebelumnya. Kode untuk melakukan pembuatan dictionary ditunjukkan pada Kode 5.11

```
dictionary = corpora.Dictionary(hasil)
dictionary.save('dictionary/dictionary.dict')
```

Kode 5.11: Pengkodean pembuatan dictionary

Fungsi *dictionary()* sendiri adalah untuk memberikan nilai unik atau sebuah *index* berupa integer pada setiap kata, yang bertujuan untuk mempermudah proses pada tahapan selanjutnya. Hasil dari dictionary yang telah diproses kemudian disimpan ke dalam file bernama *dictionary.dict*. Untuk dapat mengetahui isi dan melakukan pengecekan pada *dictionary* yang telah terbentuk dapat dilakukan fungsi cetak *dictionary* seperti pada Kode 5.12.

```
print(dictionary)
print(dictionary.token2id)
```

Kode 5.12: Pengkodean untuk mencetak *dictionary* dan jumlahnya

Melalui tahap pembuatan *dictionary*, langkah yang harus dilakukan selanjutnya adalah membuat korpus atau kumpulan vektor kata dari dokumen. Proses membuat kata menjadi korpus dapat dilakukan dengan menggunakan fungsi *doc2bow*. Kode 5.13 menjelaskan cara pembuatan korpus.

```
corpus = [dictionary.doc2bow(text) for text in
    tokens] corpora.MmCorpus.serialize('corpora.
    mm', corpus)
```

Kode 5.13: Pengkodean untuk membuat *corpus*

Hasil pembuatan korpus perlu disimpan untuk memudahkan proses pembuatan dan penggunaan model lebih lanjut. Korpus kemudian di proses untuk digunakan membuat model LDA yang paling sesuai. Model dianggap memiliki kesesuaian yang besar atau baik, dilihat dari nilai perplexity yang dimilikinya. Semakin kecil semakin baik model tersebut.

5.6 Pemodelan Topik dengan Latent Dirichlet Allocation

Tahapan dalam melakukan pemodelan topik dengan Latent Dirichlet Allocation merupakan tahapan yang dilakukan untuk membentuk model topik. Dua hal penting yang perlu diingat pada proses ini, yaitu alur pemodelan topik dan uji coba pemodelan topik.

5.6.1 Alur Pemodelan Topik menggunakan *Latent Dirichlet Allocation*

1. **Memuat *Dictionary* dan *Corpus***

   Merupakan tahapan memuat data *dictionary* dan *corpus* yang sebelumnya telah disimpan sebagai *file*. Untuk memudahkan proses memuat data, *dictionary* dan *corpus* disimpan menggunakan variabel penamaan tertentu. Pengkodean yang digunakan untuk melakukan proses memuat *dictionary* dan *Corpus* ditunjukkan pada Kode 5.14

```
dictionary = dictionary.load('dictionary/
    dictionary.dict')
corpus = [dictionary.doc2bow(text) for text
     in hasil]
corpora.MmCorpus.serialize('dictionary/
    corpora.mm', corpus)
```

Kode 5.14: *Loading dictionary* dan *Corpus*

2. **Pembentukan Model Topik**
   Pada tahapan pembentukan model topik, digunakan *library gensim* dengan *modul MmCorpus* dan *Dictionary*. Dalam pembentukan model topik, diperlukan sebuah *parameter*, yaitu berupa jumlah topik, dan *passes*. *Passes* merupakan jumlah iterasi yang dilakukan pada proses pembentukan model

topik. *Penentuan parameter* disini dimaksudkan untuk mencari nilai *perplexity* yang optimal. Semakin kecil nilai *perplexity* menunjukkan bahwa model yang dibentuk semakin baik. Adapun kode untuk melakukan uji coba *input parameter LDA* ditunjukkan pada Kode 5.16

```
from gensim import corpora, models,
    similarities
from gensim.models import ldamodel
for i in range(30):
lda_a = ldamodel.LdaModel(corpus, id2word=
    dictionary, num_topics=2, passes=15,
    alpha='auto')
lda_a.save('model/stem/model_2_30_stem.
    model')
```

Kode 5.15: uji coba *input parameter LDA*

3. **Pendokumentasian *Logging***
   Dalam melakukan uji coba, diperlukan proses pencatatan atau *logging*. Proses pencatatan dianggap penting karena berguna untuk mengetahui rekaman catatan terkait proses yang terjadi dalam proses pembentukan model topik. Informasi penting yang perlu diperhatikan adalah nilai *perplexity*. Nilai perplexity yang muncul merupakan hasil akumulasi secara otomatis oleh fitur *modul genism*. Dalam melakukan pencatatan hasil uji coba dibutuhkan sebuah *library* yang bernama *logging*, kemudian *file logging* disimpan dengan *format.csv* untuk dianalisa. Pengkodean untuk melakukan *logging* ditunjukkan dengan Kode **??**

```
import logging
import gensim

logging.basicConfig(filename='rekap_model/
    rekapmodel_2_30_stem.csv', filemode='w'
    ,format='%(asctime)s : %(levelname)s : 
    %(message)s', level=logging.INFO)
#logging.basicConfig(format='%(asctime)s :
```

```
        %(levelname)s : %(message)s', level=
        logging.INFO)
from gensim import corpora, models,
    similarities
from gensim.models import ldamodel
for i in range(30):
    lda_a = ldamodel.LdaModel(corpus,
        id2word=dictionary, num_topics=2,
        passes=15, alpha='auto')
    lda_a.save('model/stem/model_2_30_stem.
        model')
```

Kode 5.16: uji coba *input parameter LDA*

Format penamaan filename dimaksudkan untuk merepesentasikan bahwa file berisi data rekaman hasil uji coba dengan menggunakan *input parameter* jumlah topik (2), jumlah *passes* (30), dan data telah melalui proses *stemming*

5.6.2 Uji Coba Pemodelan Topik menggunakan *LDA*

Tahap uji coba pemodelan topik menggunakan Latent Dirichlet Allocation merupakan tahapan yang dilakukan untuk membentuk model topik terbaik dengan melakukan uji coba sesuai *parameter*, yakni *passes* atau jumlah iterasi dan jumlah topik.

1. **Penentuan Jumlah Iterasi**
   Pada metode *Latent Dirichlet Allocation* atau *LDA*, istilah iterasi dikenal dengan *passes*. Penentuan iterasi merupakan tahap yang penting dalam menentukan model, guna menghasilkan model yang terbaik. Apabila jumlah iterasi terlalu sedikit, akan menghasilkan model yang belum stabil atau dapat dikatakan *under fitting*, sementara iterasi yang terlalu banyak akan menghasilkan model yang *overfitting*. Penentuan jumlah *passes* diawali dengan memberikan nilai sebesar 30, kemudian jumlah topik ditentukan pada rentang 2, 3, 4, dan

5 topik. Berdasarkan hasil uji coba jumlah iterasi, nilai perplexity yang muncul akan dicatat untuk dianalisis grafiknya secara visual, dan dilakukan penghitungan standar deviasinya. Nilai passes yang akan digunakan adalah nilai passes paling awal setelah menunjukkan tren yang stabil. Adapun pengkodean yang digunakan untuk penentuan jumlah iterasi ditunjukkan dengan Kode 5.17

```
import gensim

from gensim import corpora, models, similarities
from gensim.models import ldamodel
for i in range(30):
    lda_a = ldamodel.LdaModel(corpus, id2word=dictionary,
        num_topics=2, passes=15, alpha='auto')
```

Kode 5.17: Penentuan jumlah iterasi

2. **Penentuan Jumlah Topik**
   Setelah menentukan jumlah iterasi, uji coba dilakukan dengan penentuan jumlah topik. Uji coba jumlah topik merupakan tahap yang penting dalam menentukan model, hal ini untuk menghasilkan model yang terbaik, model dapat dikatakan terbaik apabila model memiliki nilai *perplexity* yang rendah, semakin rendah nilai *perplexity*, menunjukkan akurasi model yang semakin baik. Penentuan jumlah topik dilakukan dengan melakukan uji coba terhadap jumlah topik, penentuan jumlah topik ditentukan pada rentang 2, 3, 4, dan 5 topik. Berdasarkan eksperimen jumlah topik, nilai perplexity yang muncul akan dicatat untuk dianalisis tren nilainya secara visual dan dilakukan penghitungan standar deviasinya. Sehingga pada akhirnya jumlah topik yang dipilih adalah jumlah topik yang memiliki nilai rata-rata *perplexity* paling rendah dengan standar deviasi minimum. Adapun pengkodean yang digunakan untuk penentuan jumlah topik ditunjukkan dengan Kode 5.18

```
import gensim
```

```
from gensim import corpora, models, similarities
from gensim.models import ldamodel
for i in range(30):
    lda_a = ldamodel.LdaModel(corpus, id2word=dictionary,
          num_topics=2, passes=15, alpha='auto')
```

Kode 5.18: Penentuan jumlah topik

Untuk memudahkan proses rekap data *logging* baik dalam menentukan iterasi dan topik digunakan fitur regex dengan tujuan untuk mendapatkan nilai *perplexity* yang lebih cepat dan akurat. Adapun pengkodean yang digunakan untuk proses mendapatkan nilai *perplexity* ditunjukkan dengan Kode 5.19

```
([0-9])+\.([0-9]) (?=perplexity)
([0-9] - )
```

Kode 5.19: Pengambilan nilai *perplexity* dengan *regex*

5.6.3 Menyimpan Model

Model dengan jumlah iterasi dan jumlah topik yang dianggap terbaik perlu disimpan agar dapat digunakan kembali dengan cepat. Model disimpan dalam format .model. Adapun cara menyimpan model ditunjukkan dengan Kode 5.20

```
lda_a = ldamodel.LdaModel(corpus, id2word=dictionary,
      num_topics=2, passes=15, alpha='auto')
lda_a.save('model/stem/model_2_30_stem.model')
```

Kode 5.20: Penentuan jumlah topik

Format penamaan model yang digunakan dalam penelitian ini adalah merepresentasikan jumlah topik, jumlah iterasi dan model yang melalui tahap stemming.

5.6.4 Validasi Model Topi

Validasi model topik disini mengacu pada nilai *perplexity* yang didapatkan dari hasil uji coba jumlah iterasi atau *passes*

dan jumlah topik. Model dapat dikatakan terbaik jika memiliki nilai *perplexity yang rendah*. Nilai perplexity akan dicatat untuk dianalisis tren nilainya secara visual dan dilakukan penghitungan standar deviasinya. Sehingga pada akhirnya jumlah iterasi dan jumlah topik yang dipilih adalah yang memiliki nilai rata-rata paling rendah dengan standar deviasi minimum.

5.7 Analisa Topik

Analisis topik merupakan tahapan yang dilakukan berdasarkan luaran dari LDA Model yang telah dipilih. Analisis topik dilakukan dengan mengeluarkan semua kemungkinan topik serta distribusi kata-kata dalam topik tersebut. Untuk menunjukkan hasil topik, dapat menggunakan Kode 5.21

```
import csv
from gensim.models import LdaModel
loading = LdaModel.load('model/stem/model_2_30_stem.model')
loadingcetak= loading.print_topics(num_topics=2, num_words
    =15)

print (loadingcetak)
```

Kode 5.21: Analisa Topik

Adapun luaran analisa topik ini ditunjukkan dengan tabel berupa daftar kata untuk masing-masing topik, dalam menentukan kategori topik yang digunakan acuan studi literatur [11].

5.8 Klasifikasi Data

Topik yang telah didefinisikan pada tahapan analisis topik digunakan sebagai input untuk melakukan klasifikasi. Klasifikasi data dalam proses ini merupakan tahap pelabelan dokumen dengan topik-topik yang telah ditentukan. Klasifikasi dilakukan pada keseluruhan dokumen. Untuk melakukan klasifikasi terhadap data, dapat dilakukab dengan menggunakan

Kode 5.22.

```
dictionary = corpora.Dictionary(hasil)
dictionary.load('dictionary/dictionary.dict')
# print(dictionary.token2id)

loading = LdaModel.load('model/stem/model_2_30_stem.model')
# print(loading.print_topics(num_topics=3, num_words=3))
def pre_new(doc):
    one = doc.split(' ')
    two = dictionary.doc2bow(one)
    return two

belong = loading[(pre_new(text))]
print(max(belong, key=lambda item: item[1])[0])
```

Kode 5.22: Klasifikasi

5.9 Integrasi *PHP* dengan *Python*

Sebelum melakukan visualisasi data, tahapan yang perlu dilakukan adalah menghubungkan *PHP* dengan model *python* yang telah dibuat. Hal ini bertujuan untuk proses kelanjutan program berikutnya, jika terdapat masukan berupa tambah *caption* program dapat secara otomatis melakukan pemodelan dan mengklasifikasi data. Kode *PHP* dibuat untuk menjalankan script Python untuk melakukan pemodelan topik dan melakukan klasifikasi topik yang akan kembali disimpan ke dalam database. Untuk mengintegrasikan *PHP* dengan *Python* dapat menggunakan Kode 5.23

```
<?php
include( 'views/header2.php' );
$topik2 = shell_exec('python_prepocessing_with_stem.py');
?>
```

Kode 5.23: Integrasi *PHP* dengan *Python* pada proses modeling data

```
<?php
 include( 'views/header2.php' );
 $db->pfx='';
 ini_set('max_execution_time', 0);
```

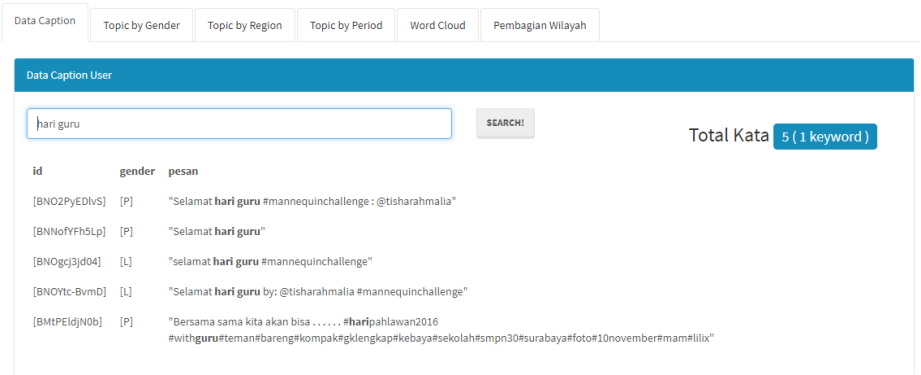
```
$f2 = $db->r('caption_bot', 'id, pesan', 'WHERE
      caption_bot.id NOT IN (SELECT id_caption FROM
      cluster_2)');
         foreach ($f2 as $r2) {
                  $id=$r2['id'];
                  $pesan=$r2['pesan'];
                  $result = shell_exec('python clasify2.py
                        "' .$pesan.'"');
         $db->i( 'cluster_2', 'id_caption, topik', array
               ($id, $result));
}
?>
```

Kode 5.24: Integrasi *PHP* dengan *Python* pada proses klasifikasi data

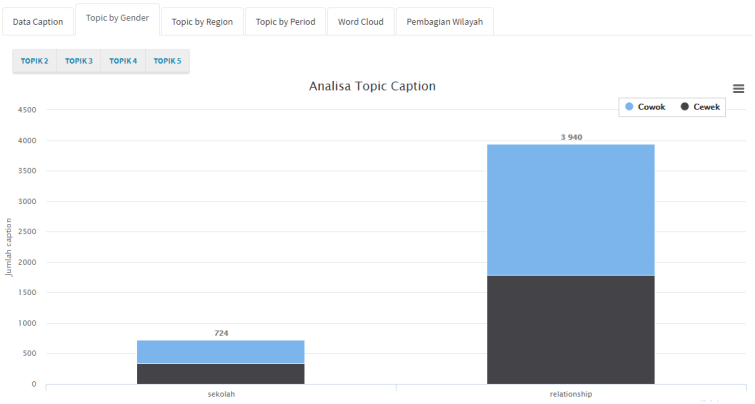
5.10 Visualisasi Data

Visualisasi data dalam penelitian ini merupakan pembuatan dashboard yang dihasilkan dari data *caption user* siswa SMP di Surabaya yang masuk melalui aplikasi menggunakan proses *crawling*. *Dashboard* dibuat untuk menunjukkan trend topic yang sering dibicarakan oleh siswa SMP di Surabaya. Visualisasi data ke dalam *dashboard* dibuat dengan menggunakan bahasa pemrograman *PHP*. Adapun bentuk visualisasi data menggunakan grafik dan tabel. Grafik ditampilkan dengan menggunakan *bar chart* dengan menggunakan *library highchart* dan visualisasi berupa tabel menggunakan *template bootstrap*. Berikut ini adalah beberapa tampilan antarmuka apliasi dan visualisasi data:

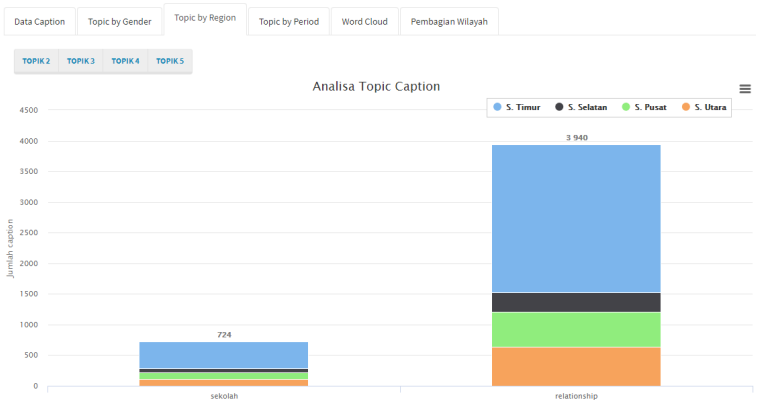
1. **Antar muka pencarian caption berdasarkan *keyword* tertentu** Fitur pencarian data *caption* berfungsi untuk mencari data berupa *caption* berdasarkan *input keyword* tertentu dari *user*, selanjutnya sistem akan menampilkan *caption* sesuai dengan *keyword* yang dimasukkan.
2. **Antar muka *dashboard* topik model berdasarkan gender** Fitur *Dashboard topic by gender* menampilkan distribusi gen-



Gambar 5.2: Antar muka pencarian caption berdasarkan *keyword* tertentu



Gambar 5.3: Antar muka *dashboard* topik model berdasarkan gender

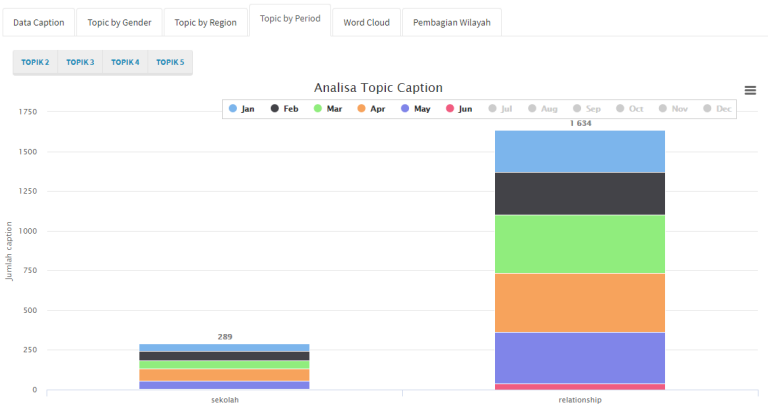


Gambar 5.4: Antar muka *dashboard* topik model berdasarkan region

der ke dalam kategori berdasarkan model topik.

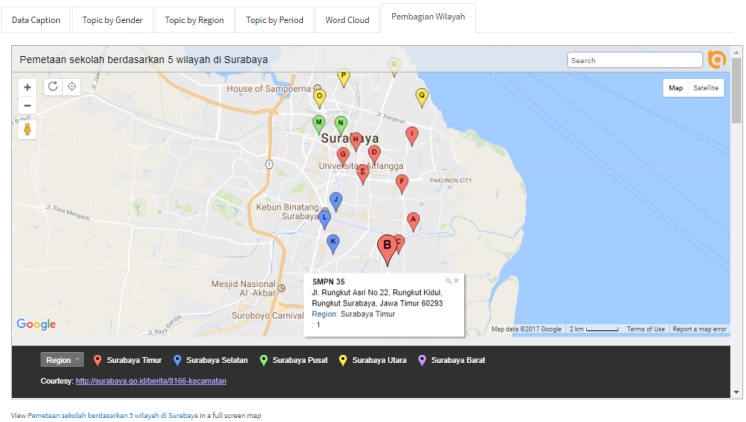
3. **Antar muka *dashboard* topik model berdasarkan region** Fitur *Dashboard topic by region* menampilkan distribusi *region* atau wilayah ke dalam kategori berdasarkan model topik.
4. **Antar muka *dashboard* topik model berdasarkan periode**

   Fitur *Dashboard topic by period* menampilkan distribusi periode ke dalam kategori berdasarkan model topik.
5. **Antar muka visualisasi wordcloud** Fitur visualisasi *wordcloud* menampilkan kata-kata yang mayoritas muncul dalam *caption instagram* siswa SMP.
6. **Antar muka visualisasi pembagian wilayah** Fitur visualisasi pembagian wilayah menampilkan lokasi sekolah yang menjadi sasaran sosialisasi mata kuliah Etika Profesi.



Gambar 5.5: Antar muka pencarian caption berdasarkan periode

Gambar 5.6: Antar muka visualisasi wordcloud



Gambar 5.7: Antar muka visualisasi pembagian wilayah

Halaman ini sengaja dikosongkan

BAB 6

HASIL DAN PEMBAHASAN

Pada bab ini akan dijelaskan terkait analisa dan pengujian yang meliputi tiga hal, yaitu analisa hasil pemodelan, pengujian fungsional dan non fungsional

6.1 Analisa Hasil Pemodelan

Pada bab analisa hasil pemodelan akan membahas mengenai hasil pengujian pembuatan model menggunakan *perplexity*. Adapun langkah-langkah yang dilakukan dijelaskan pada lebih lanjut.

6.1.1 Memuat Data

Total data yang berhasil didapatkan dari hasil *crawling* berjumlah 4664 data. Adapun detail data yang berhasil terakuisisi ditunjukkan dengan Tabel 6.1. Berdasarkan data Tabel 6.1, dapat dikatakan bahwa secara keseluruhan terjadi peningkatan intensitas aktivitas *posting* oleh pengguna *instagram* setiap tahunnya. Pada kolom 2017 di bulan Juli sampai dengan Desember didapatkan nilai 0, dikarenakan proses pengambilan data dilakukan pada bulan Januari hingga Juni.

Tabel 6.1: Jumlah data *caption* siswa SMP di Surabaya

Bulan	2014	2015	2016	2017
Januari	0	17	60	308
Februari	0	10	71	332
Maret	1	11	87	418
April	0	16	106	447
Mei	0	15	150	378
Juni	4	44	186	40
Juli	3	35	285	0
Agustus	3	33	257	0
September	5	47	201	0
Oktober	21	37	219	0
November	22	54	295	0
Desember	24	69	353	0
Jumlah	**83**	388	2270	1923
Total Data	**4664**			
Total Kata	22083			

6.1.2 Pra-Proses Data

Pendefinisian *Data Training*

Dalam penelitian ini untuk membuat model berdasarkan data *caption instagram* siswa SMP, digunakan data sebanyak (70 persen) dari total data 4664, yakni 3265 data.

Pendefinisian *Stopword*

Stopwords merupakan kata umum yang biasanya muncul dalam jumlah besar dianggap tidak memiliki makna. Dalam penelitian ini, pendefinisian stopword dilakukan dengan dua landasan asumsi, yaitu:

1. Berdasarkan analisa intensitas kemunculan kata dalam (5000 kata tertinggi)

Berdasarkan pemaknaan kata sesuai sistem tata Bahasa Indonesia baku [1]

Dalam tata Bahasa Indonesia baku, terdapat istilah kelas kata, yaitu istilah golongan kata berdasarkan bentuk, fungsi, dan maknanya (KBBI). Dari berbagai kelas kata yang ada, terdapat beberapa kelas kata yang jika berdiri sendiri tanpa disertai dengan kata yang diterangkan, kata tersebut tidak memiliki makna, sehingga kelas kata yang dianggap memenuhi syarat untuk digolongkan ke dalam stopword dapat dilihat pada Tabel berikut:

Adapun hasil dari proses *stopword removal* ditunjukkan dengan Tabel 6.7

Tabel 6.2: Kata depan sebagai kata tugas

Kata Depan (Preposisi)			
Tempat	Maksud	Waktu	Sebab
di	untuk	hingga	demi
ke	guna	hampir	atas
dari	agar	nyaris	karena

Tabel 6.3: Kata sambung sebagai kata tugas

Kata Sambung			
Asal	**Jadian / bentukan**	**Majemuk**	**Berimbuhan**
dan	jangan-jangan	apabila	sebelum
maka	seakan-akan	lagi pula	selama
sedang	kalau-kalau	karena itu	sehingga

Tabel 6.4: Kata seru sebagai kata tugas

Kata Seru (Interjeksi)	
Singkat	Biasa
wah	aduh
cih	celaka
hai	ya ampun

Tabel 6.5: Kata sandang sebagai kata tugas

Kata Sandang			
Jumlah tunggal	**Jamak/kelompok**	**Kata ganti orang/ benda**	**Berimbuhan**
dan	jangan-jangan	dia	sebelum
sang	para	si	selama

Tabel 6.6: Partikel penegas sebagai kata tugas

Partikel Penegas
Jumlah tunggal
kah
lah
tah

Tabel 6.7: Hasil pra-proses data menggunakan *stopword*

	Jumlah Kata
Sebelum	**22083**
Setelah	**11178**

6.1.3 Pembentukan Model Topik dengan LDA

Dalam melakukan pembentukan model LDA, model yang melalui dua tahap uji coba, yakni melalui tahap *stemming* dan tanpa melalui tahap *stemming*

Pembentukan Model *LDA* dengan *stemming*

Distribusi probabilitas kata dalam 2 Topik dengan 15 kata per topik dari hasil model yang melalui proses *stemming* dapat dilihat pada Tabel 6.8.

Pembentukan Model *LDA* tanpa *stemming*

Distribusi probabilitas kata dalam 2 Topik dengan 15 kata per topik dari hasil model yang tanpa melalui proses *stemming* dapat dilihat pada Tabel 6.9

Tabel 6.8: Hasil Pembentukan Model LDA dengan Stemming

2 Topic	
Topik 0	Topik 1
0.010*hati	0.009*sukses
0.010*surabaya	0.007*selfie
0.009*sayang	0.007*sahabat
0.008*selamat	0.007*kawan
0.007*senyum	0.005*temen
0.007*photograph	0.004*rindu
0.006*kelas	0.004*sehat
0.006*hbd	0.004*usaha
0.005*tuhan	0.004*challenge
0.005*juang	0.003*semangat
0.005*indonesia	0.003*makan
0.004*kangen	0.003*surabaya
0.004*percaya	0.003*nakal
0.004*lomba	0.003*smp
0.004*family	0.003*mannequinchallenge

Tabel 6.9: Hasil Pembentukan Model LDA dengan Stemming

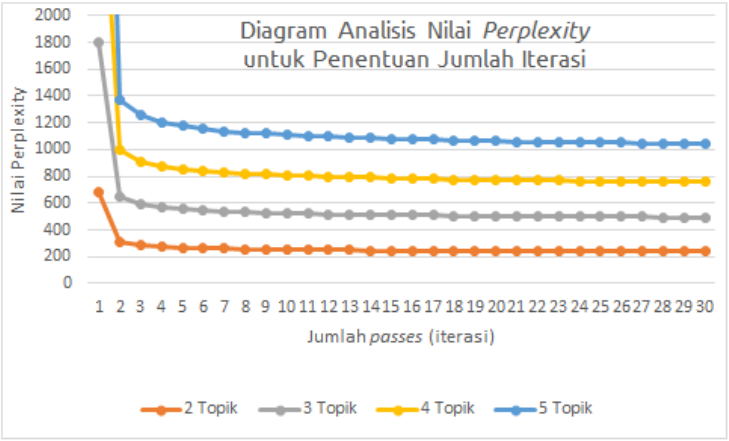
2 Topic	
Topik 0	Topik 1
0.005*selamat	0.004*bahagia
0.004*selfie	0.003*jalan
0.004*sukses	0.003*sahabat
0.004*surabaya	0.002*miss
0.004*kawan	0.002*challenge
0.004*hati	0.002*drummerindonesia
0.004*hidup	0.002*car
0.003*kelas	0.002*sederhana
0.003*photograph	0.002*satriaandthemonster
0.003*sayang	0.002*smile
0.003*tuhan	0.002*nissan
0.003*canon	0.002*berjuang
0.003*hbd	0.002*kebersamaan
0.003*kakak	0.002*carxdriftracing
0.002*eos	0.001*accepted

6.1.4 Validasi Model Topik

Validasi model topik pada penelitian ini mengacu pada nilai *perplexity* dari proses penentuan iterasi dan penentuan jumlah topik.

Penentuan Jumlah Iterasi

Dalam melakukan penentuan jumlah iterasi atau *passes*, metode yang digunakan adalah dengan menganalisa nilai *perplexity*. Analisis nilai perplexity dalam penelitian ini dilakukan dengan cara menjalankan pemodelan terhadap topik, dengan parameter jumlah topik yang dipilih adalah 2, 3, 4 dan 5. Hasil dari nilai perplexity yang muncul dari masing-masing parameter jumlah topik tersebut kemudian dicatat dan divisualisasikan pada Gambar 6.1



Gambar 6.1: Analisa nilai *perplexity* untuk penentuan jumlah iterasi

Berdasarkan Gambar 6.1, dapat dilihat bahwa nilai perplexity sudah mencapai kondisi cenderung stabil pada passes ke 15 untuk keseluruhan parameter jumlah topik. Maka dapat disimpulkan bahwa iterasi yang digunakan adalah 15. Hal ini juga didukung oleh Gambar 6.2 yang menunjukkan selisih nilai *perplexity* pada iterasi ke 15 sudah mencapai kondisi konstan.

Hasil Percobaan 1 Perplexity regular Stopword pass : 30 With_Stem								
Passes	2 Topik		3 Topik		4 Topik		5 Topik	
1	683	Selisih	1121	Selisih	1713,7	Selisih	2662,6	Selisih
2	306,6	376,4	349,8	771,2	353,3	1360,4	383,7	2278,9
3	282,9	23,7	312,8	37	324,5	28,8	347,6	36,1
4	272,1	10,8	297,5	15,3	311,9	12,6	331,4	16,2
5	265,8	6,3	288,9	8,6	304,6	7,3	322,8	8,6
6	261,5	4,3	283,1	5,8	299	5,6	316,6	6,2
7	258,5	3	279,2	3,9	294,2	4,8	311,8	4,8
8	256,2	2,3	276	3,2	290,8	3,4	308,3	3,5
9	254,4	1,8	273,5	2,5	287,7	3,1	305,2	3,1
10	252,9	1,5	271,5	2	284,2	3,5	302,6	2,6
11	251,6	1,3	269,9	1,6	281,6	2,6	300,4	2,2
12	250,5	1,1	268,5	1,4	279,5	2,1	298,7	1,7
13	249,5	1	267,2	1,3	278	1,5	296,9	1,8
14	248,7	0,8	266,1	1,1	276,7	1,3	295,5	1,4
15	247,9	0,8	265	1,1	275,4	1,3	294,3	1,2
16	247,2	0,7	264,1	0,9	274,3	1,1	293,1	1,2
17	246,6	0,6	263,2	0,9	273,3	1	292,3	0,8
18	246	0,6	262,4	0,8	272,5	0,8	291,5	0,8
19	245,4	0,6	261,7	0,7	271,7	0,8	290,8	0,7
20	244,9	0,5	260,9	0,8	270,9	0,8	290	0,8
21	244,4	0,5	260,1	0,8	270,1	0,8	289,4	0,6

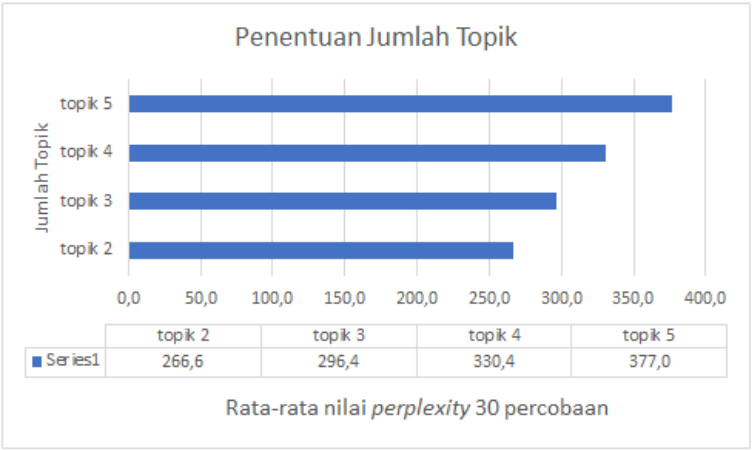
Gambar 6.2: Analisa nilai *perplexity* untuk penentuan jumlah iterasi berdasarkan selisih nilai *perplexity*

Penentuan Jumlah Topik

Pada tahap penentuan jumlah topik, dilakukan dengan menganalisis nilai *perplexity*, dengan konteks untuk penentuan jumlah topik. Tahap ini dilakukan dengan cara uji coba parameter jumlah topik dengan rentang nilai yang lebih luas, dalam hal ini rentang nilai yang digunakan untuk uji coba ditampilkan dalam gambar 6.3. Analisis nilai perplexity dalam konteks untuk penentuan jumlah topik dilakukan dengan melakukan *running* sebanyak 30 kali untuk mendapatkan rata-rata nilai perplexity yang akurat untuk masing-masing topik.

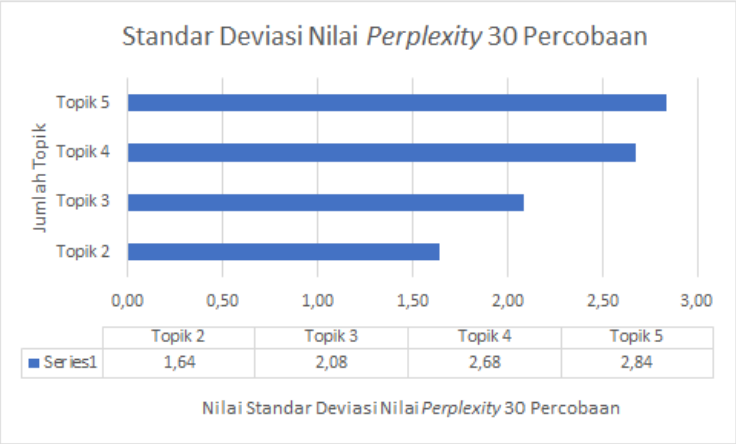
Penentuan Jumlah Topik dengan *Stemming*

Hasil uji coba analisis nilai *perplexity* untuk penentuan jumlah topik dengan *stemming* dapat dilihat pada lampiran. Berdasarkan Gam-



Gambar 6.3: Rata-rata nilai *perplexity* 30 percobaan

bar 6.3, nilai perplexity terendah terdapat pada jumlah topik 2 yaitu 266.6, dan nilai *perplexity* meningkat untuk jumlah topik yang semakin tinggi, sehingga 2 topik merupakan jumlah topik terbaik berdasarkan analisis nilai *perplexity*.

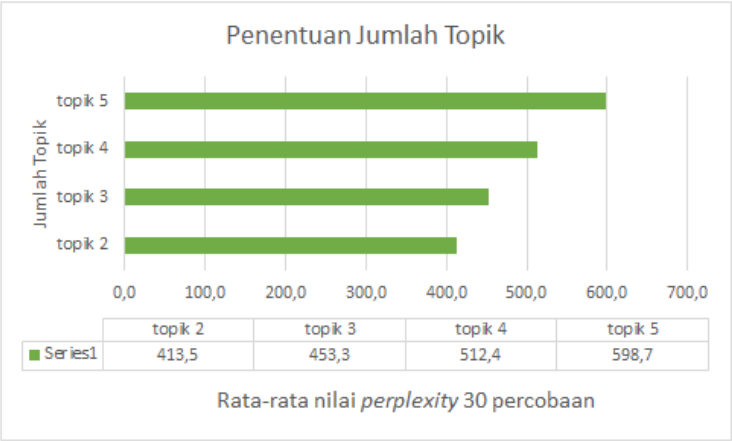


Gambar 6.4: Standar Deviasi Nilai *Perplexity* 30 Percobaan

Berdasarkan Gambar 6.4, diketahui bahwa standar deviasi nilai *perplexity* 2 topik untuk 30 kali percoban adalah 1.64. jika dibandingkan dengan nilai rata-rata standar deviasi keseluruhan yaitu 2.31.

Penentuan Jumlah Topik tanpa *Stemming*

Hasil uji coba analisis nilai *perplexity* untuk penentuan jumlah topik tanpa *stemming* dapat dilihat pada lampiran. Berdasarkan Gambar 6.5, nilai perplexity terendah terdapat pada jumlah topik 2 yaitu 413,5, dan nilai *perplexity* meningkat untuk jumlah topik yang semakin tinggi, sehingga 2 topik merupakan jumlah topik terba-

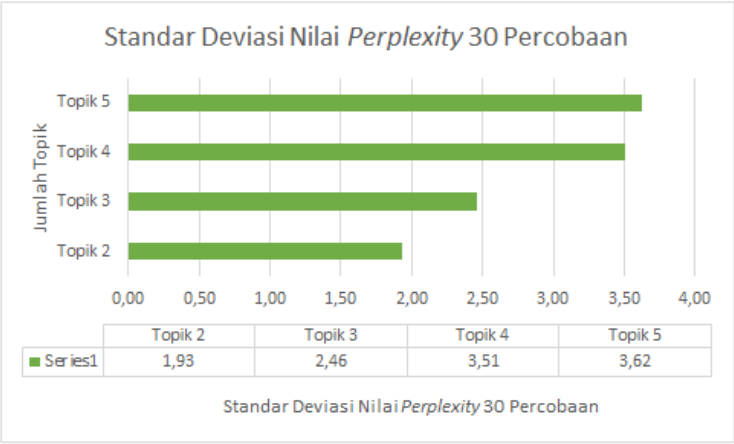


Gambar 6.5: Rata-rata nilai *perplexity* 30 percobaan

ik berdasarkan analisis nilai *perplexity*. Berdasarkan Gambar 6.6, diketahui bahwa standar deviasi nilai *perplexity* 2 topik untuk 30 kali percoban adalah 1.93, jika dibandingkan dengan nilai rata-rata standar deviasi keseluruhan yaitu 2.88. Melalui tahap validasi berdasarkan nilai *perplexity* untuk menentukan jumlah iterasi dan jumlah topik, dapat dikatakan bahwa topik 2 merupakan topik terbaik sesuai dengan nilai *perplexity* yang didapatkan.

6.2 Pengujian Fungsional

Untuk Pengujian Fungsionalitas dari aplikasi, dengan melakukan berbagai skenario penggunaan aplikasi dimana setiap skenario menguji fungsionalitas berbeda dari aplikasi. Adapun pengujian yang akan dilakukan yakni, pertama melakukan pengujian Fungsionalitas pada fitur penambahan data *caption* melalui proses *crawling* yang

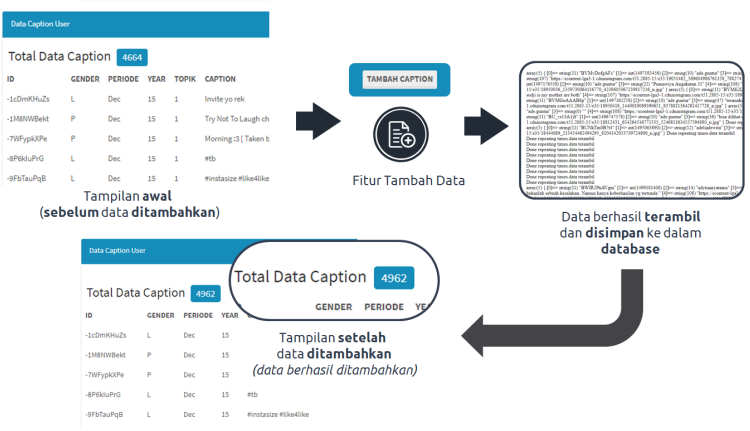


Gambar 6.6: Standar Deviasi Nilai Perplexity 30 Percobaan

disediakan pada tombol tambah *caption*. kedua melakukan pengujian pada fitur pembuatan model dengan cara mengeksekusi *script python* melalui *PHP*, dan ketiga pengujian pada fitur prediksi label data dengan cara mengeksekusi *script python* melalui *PHP* untuk menjalankan pelabelan data *caption*.

6.2.1 Fitur Tambah Data

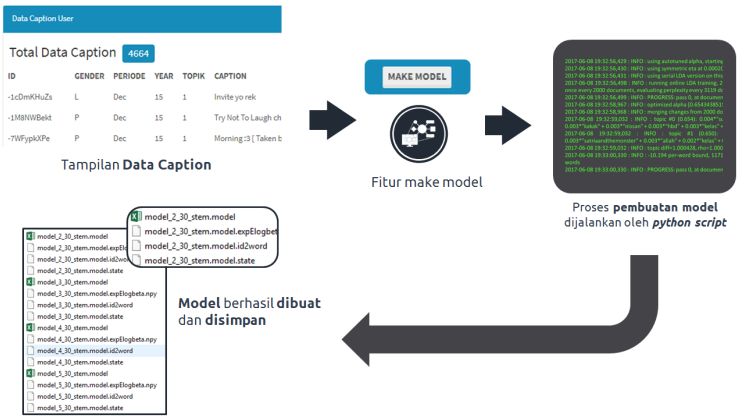
Pada fitur tambah data disediakan fungsi untuk melakukan *crawling* data yang bertujuan untuk menambahkan data *caption* ke dalam database. Gambar 6.7 adalah tampilan dari fitur tambah data ketika dijalankan dan respon yang diberikan oleh sistem. Berdasarkan Gambar 6.7 menunjukkan proses penambahan data berhasil dilakukan, dibuktikan dengan bertambahnya data caption dari yang semula **4664 data** menjadi **4962 data**.



Gambar 6.7: Fitur penambahan data *caption* melalui proses *crawling*

6.2.2 Fitur Membuat Model

Pada fitur membuat model disediakan fungsi untuk menjalankan *script python* dengan tujuan untuk melakukan proses pembuatan model dan menyimpan model tersebut. Untuk kebutuhan aplikasi ini dibutuhkan 4 model yang kemudian dijadikan acuan untuk memvisualkan data caption menggunakan *PHP*. Gambar 6.8 adalah tampilan dari fitur *make model* ketika dijalankan dan respon sistem terhadap *action*. Proses pembuatan model telah berhasil dilakukan. Dibuktikan dengan bertambahnya data model yang akan digunakan untuk proses prediksi label data.



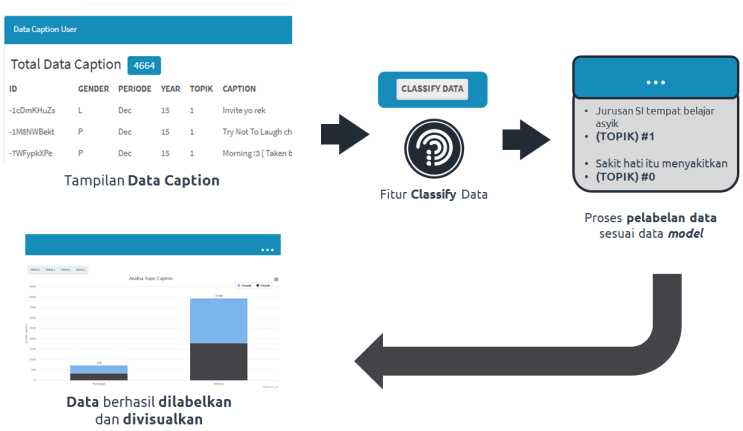
Gambar 6.8: Fitur pembuatan model menggunakan input tombol

6.2.3 Fitur Prediksi Label Data

Pada fitur prediksi label data disediakan fungsi untuk menjalankan *script python* dengan tujuan untuk melakukan proses pelabelan data. Kemudian data yang telah dilabelkan disimpan ke dalam database untuk selanjutnya divisualkan. Berdasarkan Gambar 6.9 menunjukkan proses prediksi label data berhasil dilakukan. Dibuktikan dengan bertambahnya label pada data yang barusaja ditambahkan dan berhasil divisualkan seperti pada Gambar 6.10, 6.11, dan 6.12 yang menjelaskan hasil prediksi *caption* ke dalam beberapa kategori topik, yang dibagi berdasarkan gender, regional, dan periode.

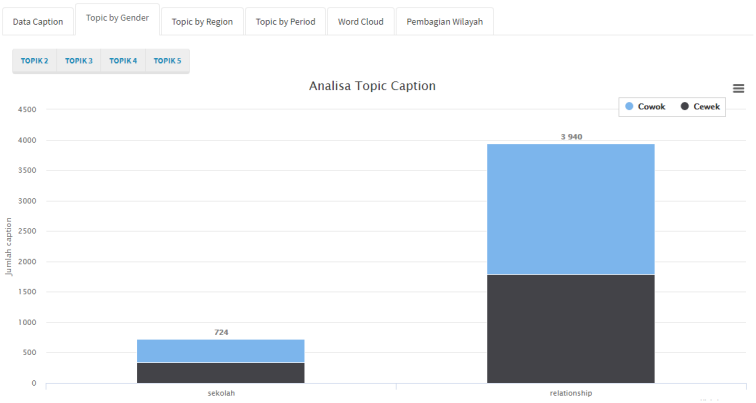
6.3 Pengujian Non Fungsional

Untuk melakukan pengujian non fungsional dari aplikasi, dilakukan pengujian berupa membandingkan tampilan aplikasi secara *web*

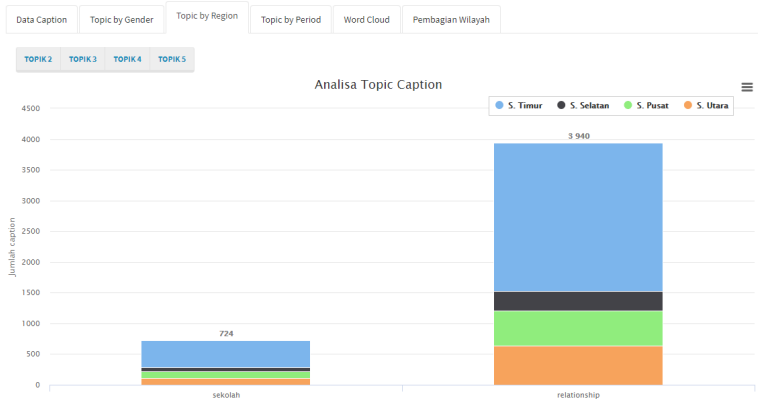


Gambar 6.9: Fitur prediksi label data

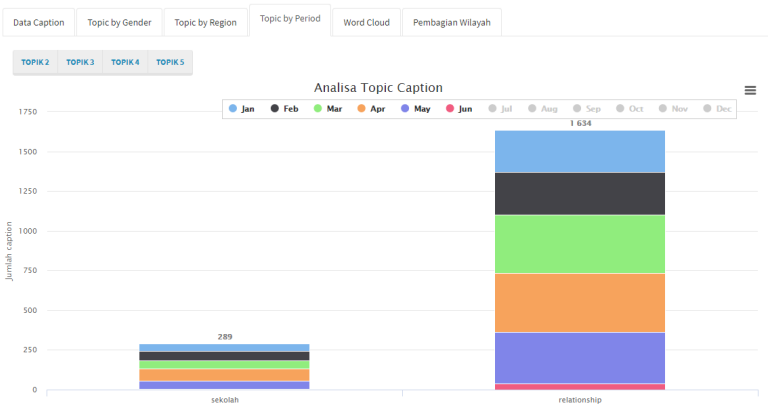
view atau tampilan web dan *mobile view*. Berdasarkan Gambar 6.13 menunjukkan contoh tampilan dapat dibuka dengan baik pada *web view* maupun *mobile view*.



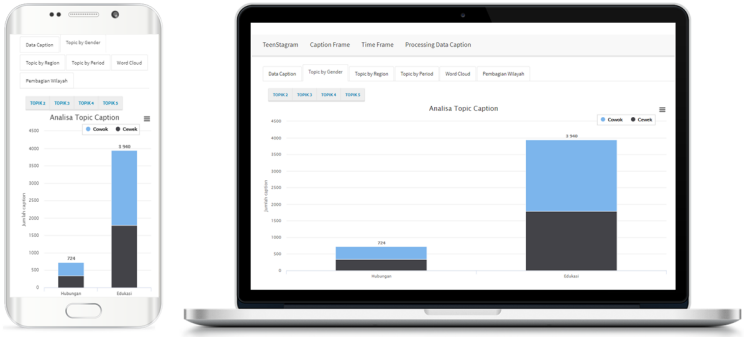
Gambar 6.10: Antar muka *dashboard* topik model berdasarkan gender



Gambar 6.11: Antar muka *dashboard* topik model berdasarkan region



Gambar 6.12: Antar muka pencarian caption berdasarkan periode



Gambar 6.13: Perbandingan tampilan secara *web view* dan *mobile view*

BAB 7

KESIMPULAN DAN SARAN

Pada bab ini akan dijelaskan kesimpulan dan saran dalam pengerjaan tugas akhir.

7.1 Kesimpulan

Berdasarkan proses-proses yang telah dilakukan dalam pengerjaan tugas akhir dengan judul ”Rancang Bangun Perangkat Lunak Teenstagram untuk Mengelompokkan Topik *Caption* Akun Instagram Siswa Sekolah Menengah Pertama Di Surabaya” yang telah dilakukan, dapat disimpulkan sebagai berikut:

1. Penelitian ini membuktikan bahwa metode *crawling* data dengan memanfaatkan *instagram API* mampu membatu proses pengumpulan data *caption* milik akun siswa SMP yang telah melalui proses validasi akun.
2. Perancangan aplikasi pada penelitian ini *TeenStagram* mengintegrasikan antara bahasa *PHP* untuk kebutuhan *crawling* dan tampilan antarmuka serta pemanfaatan *Python* untuk melakukan pembuatan model sekaligus melakukan prediksi label data, mampu menjawab kebutuhan visualisasi data *caption* akun siswa SMP.
3. Pendefinisian stopword yang dihasilkan pada penelitian dapat digunakan sebagai acuan untuk tahap praproses data pada penelitian selanjutnya dalam lingkup studi kasus sosial media remaja.

7.2 Saran

Saran penulis untuk penelitian selanjutnya sebagai berikut:

1. Adanya permasalahan terkait data teks, yaitu moyoritas struktur kalimat yang terkandung di dalam *caption* akun *instagram* siswa SMP berupa bahasa yang tidak baku. Hal ini sangat berpengaruh pada proses pendefinisian *stopword* dan pembuatan model. Oleh sebab itu akan lebih baik jika dilakukan pengamatan struktur kalimat dan kata terlebih dahulu, sebelum melakukan *topic modelling*.
2. Harapan kedepan, penelitian ini dapat dikembangkan ke arah mengidentifikasi kalimat yang mengandung konotasi negatif atau positif, sehingga dapat diambil tindakan terhadap *caption* atau informasi yang tidak bertanggung jawab.
3. Pada penelitian ini belum dilakukan *user acceptance test*. Pada penelitian selanjutnya diharapkan melakukan *user acceptance test* untuk menguji kemudahan pengguna dalam menggunakan aplikasi.

DAFTAR PUSTAKA

[1] Hasan Alwi. dkk. 1998. *Tata Bahasa Baku Bahasa Indonesia*.

[2] APJJI. Infografis penetrasi & pengguna internet indonesia., 2016.

[3] Kent Beck, Mike Beedle, Arie Van Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, James Grenning, Jim Highsmith, Andrew Hunt, Ron Jeffries, et al. Manifesto for agile software development. 2001.

[4] David M Blei. Probabilistic topic models. *Communications of the ACM*, 55(4):77–84, 2012.

[5] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.

[6] Joshua Charles Campbell, Abram Hindle, and Eleni Stroulia. Latent dirichlet allocation: extracting topics from software engineering data. *The art and science of analyzing software data*, 1, 2014.

[7] Carlos Castillo. Effective web crawling. In *ACM SIGIR Forum*, volume 39, pages 55–56. Acm, 2005.

[8] Bruce Ferwerda, Markus Schedl, and Marko Tkalcic. Using instagram picture features to predict users' personality. In *International Conference on Multimedia Modeling*, pages 850–861. Springer, 2016.

[9] Allan Heydon and Marc Najork. Mercator: A scalable, extensible web crawler. *World Wide Web*, 2(4):219–229, 1999.

[10] Homa Hosseinmardi, Sabrina Arredondo Mattson, Rahat Ibn Rafiq, Richard Han, Qin Lv, and Shivakant Mishra. Detection of cyberbullying incidents on the instagram social network. *arXiv preprint arXiv:1503.03909*, 2015.

[11] Yuheng Hu, Lydia Manikonda, Subbarao Kambhampati, et al. What we instagram: A first analysis of instagram photo content and user types. In *ICWSM*, 2014.

[12] developer Instagram. Instagram developer documentation@ONLINE, February 2017.

[13] developer Instagram. What are your limits on instagram?@ONLINE, February 2017.

[14] UNICEF KOMINFO. Riset kominfo dan unicef mengenai perilaku anak dan remaja dalam menggunakan internet @ONLINE, February 2014.

[15] Jey Han Lau, Nigel Collier, and Timothy Baldwin. On-line trend analysis with topic models:\# twitter trends detection topic model online. In *COLING*, pages 1519–1534, 2012.

[16] Manuel Ignacio Lopez, JM Luna, C Romero, and S Ventura. Classification via clustering for predicting final marks based on student participation in forums. *International Educational Data Mining Society*, 2012.

[17] Lydia Manikonda, Yuheng Hu, and Subbarao Kambhampati. Analyzing user activities, demographics, social network structure and user-generated content on instagram. *arXiv preprint arXiv:1410.8099*, 2014.

[18] Roger S Pressman. *Software engineering: a practitioner's approach*. Palgrave Macmillan, 2005.

[19] Radim Rehurek and Petr Sojka. Software framework for topic modelling with large corpora. In *In Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. Citeseer, 2010.

[20] Kristie Seymore and Roni Rosenfeld. Using story topics for language model adaptation. 1997.

[21] Astrid Kurnia Sherlyanita and Nur Aini Rakhmawati. Pengaruh dan pola aktivitas penggunaan internet serta media sosial pada siswa smpn 52 surabaya. *Journal of Information Systems Engineering and Business Intelligence*, 2(1):17–22, 2016.

[22] Stefan Stieglitz and Linh Dang-Xuan. Social media and political communication: a social media analytics framework. *Social Network Analysis and Mining*, 3(4):1277–1291, 2013.

[23] Zhou Tong and Haiyi Zhang. A text mining research based on lda topic modelling. *A TEXT MINING RESEARCH BASED ON LDA TOPIC MODELLING*, 10.5121/csit.2016.60616:201–210, 2016.

[24] Jui-Feng Yeh, Yi-Shan Tan, and Chen-Hsien Lee. Topic detection and tracking for conversational content by using conceptual dynamic latent dirichlet allocation. *Neurocomputing*, 216:310–318, 2016.

Halaman ini sengaja dikosongkan

BIODATA PENULIS

Penulis lahir di Kota Jamu, Wonogiri pada tanggal 11 November 1994. Merupakan anak pertama dan satu-satunya yang telah menempuh pendidikan formal yaitu; SD Negeri Badas II Pare, SMP Negeri 2 Pare, dan SMA Negeri 2 Pare.

Pada tahun 2013 melanjutkan pendidikan di Jurusan Sistem Informasi FTIF - Institut Teknologi Sepuluh Nopember (ITS) Surabaya dan terdaftar sebagai mahasiswa dengan NRP 5213100055. Selama menjadi mahasiswa penulis telah mengikuti berbagai kegiatan kemahasiswaan, dan kompetisi baik di tingkat Institut maupun ditingkat Nasional. Disamping itu serta aktif sebagai Kepala Departemen Sosial Masyarakat di Himpunan Mahasiswa Sistem Informasi ITS 2015/2016. Disamping aktif dalam kegiatan kemahasiswaan, penulis juga pernah menjadi ketua asisten dosen pada matakuliah Pengantar Sistem Operasi dan asisten praktikum pada mata kuliah Interaksi Manusia dan Komputer.

Pada tahun keempat karena penulis tertarik dengan bidang *social network analysist*, sehingga mengambil bidang minat Laboratorium Akuisisi Data dan Diseminasi Informasi (ADDI). Penulis dapat dihubungi melalui email tetha13@mhs.is.its.ac.id.